vSphere Security

Update 3
Modified on 08 MAR 2024
VMware vSphere 7.0
VMware ESXi 7.0
vCenter Server 7.0

vmware
by Broadcom

You can find the most up-to-date technical documentation on the VMware by Broadcom website at:

https://docs.vmware.com/

VMware by Broadcom
3401 Hillview Ave.
Palo Alto, CA 94304
www.vmware.com

Contents

About vSphere Security

vSphere Security provides information about securing your vSphere® environment for VMware® vCenter® Server and VMware ESXi.

At VMware, we value inclusion. To foster this principle within our customer, partner, and internal community, we create content using inclusive language.

To help you protect your vSphere environment, this documentation describes available security features and the measures that you can take to safeguard your environment from attack.

Table 1-1. *vSphere Security* Highlights

Topics	Content Highlights
Permissions and User Management	■ Permissions model (roles, groups, objects). ■ Creating custom roles. ■ Setting permissions. ■ Managing global permissions.
Host Security Features	■ Lockdown mode and other security profile features. ■ Host smart card authentication. ■ vSphere Authentication Proxy. ■ UEFI Secure Boot. ■ Trusted Platform Module (TPM). ■ VMware® vSphere Trust Authority™. ■ Secure ESXi Configuration and configuration sealing
Virtual Machine Encryption	■ VMware vSphere® Native Key Provider™. ■ How does VM encryption work? ■ KMS setup. ■ Encrypting and decrypting VMs. ■ Troubleshooting and best practices.
Guest OS Security	■ Virtual Trusted Platform Module (vTPM). ■ Virtualization Based Security (VBS).
Managing TLS Protocol Configuration	Changing TLS protocol configuration using a command-line utility.

Table 1-1. *vSphere Security* Highlights (continued)

Topics	Content Highlights
Security Best Practices and Hardening	Best practices and advice from VMware security experts. ■ vCenter Server security ■ Host security ■ Virtual machine security ■ Networking security
vSphere Privileges	Complete listing of all vSphere privileges supported in this release.

Related Documentation

A companion document, *vSphere Authentication*, explains how you can use authentication services, for example, to manage authentication with vCenter Single Sign-On and to manage certificates in your vSphere environment.

In addition to these documents, VMware publishes the *vSphere Security Configuration Guide* (formerly known as the *Hardening Guide*) for each release of vSphere, accessible at https://core.vmware.com/security. The *vSphere Security Configuration Guide* contains guidelines on security settings that can or should be set by the customer, and security settings delivered by VMware that should be audited by the customer to ensure that they are still set to default.

What Happened to the Platform Services Controller

Beginning in vSphere 7.0, deploying a new vCenter Server or upgrading to vCenter Server 7.0 requires the use of the vCenter Server appliance, a preconfigured virtual machine optimized for running vCenter Server. The new vCenter Server contains all Platform Services Controller services, preserving the functionality and workflows, including authentication, certificate management, tags, and licensing. It is no longer necessary nor possible to deploy and use an external Platform Services Controller. All Platform Services Controller services are consolidated into vCenter Server, and deployment and administration are simplified.

As these services are now part of vCenter Server, they are no longer described as a part of Platform Services Controller. In vSphere 7.0, the *vSphere Authentication* publication replaces the *Platform Services Controller Administration* publication. The new publication contains complete information about authentication and certificate management. For information about upgrading or migrating from vSphere 6.5 and 6.7 deployments using an existing external Platform Services Controller to vSphere 7.0 using vCenter Server appliance, see the *vSphere Upgrade* documentation.

Intended Audience

This information is for experienced system administrators who are familiar with virtual machine technology and data center operations.

Certifications

VMware publishes a public list of VMware products that have completed Common Criteria certifications. To check if a particular VMware product version has been certified, see the Common Criteria Evaluation and Validation webpage at https://www.vmware.com/security/certifications/common-criteria.html.

Updated Information

This *vSphere Security* document is updated with each release of the product or when necessary.

This table provides the update history of the *vSphere Security* documentation.

Revision	Description
08 MAR 2024	■ Added new topic, Delete a Standard Key Provider.
01 FEB 2024	■ Minor update to Enable or Disable UEFI Secure Boot for a Virtual Machine.
22 JAN 2024	■ Minor update to ESXi Certificate Default Settings.
10 JAN 2024	■ Minor update to Set Up vSphere Authentication Proxy to Use Custom Certificates. ■ Minor update to Decrypt an Encrypted Virtual Machine or Virtual Disk.
11 DEC 2023	■ Minor update to Unlock Locked Virtual Machines. ■ Minor update to Scan vCenter Server for Enabled TLS Protocols. ■ Minor update to vSphere with Tanzu Privileges.
17 NOV 2023	■ Updated the vSphere Inventory Hierarchy figure in Hierarchical Inheritance of Permissions. ■ Minor update to ESXi Pass Phrase. ■ Minor update to ESXi Certificate Default Settings. ■ Minor update to Add Users to the DCUI.Access Advanced Option. ■ Minor update to Enable or Disable the Secure Boot Enforcement for a Secure ESXi Configuration. ■ Minor update to Restore a vSphere Native Key Provider Using the vSphere Client. ■ Updated Create a Virtual Machine with a Virtual Trusted Platform Module and Enable Virtual Trusted Platform Module for an Existing Virtual Machine to state that the **Cryptographic operations.Direct Access** privilege is required to open a console session. ■ Minor update to Enable or Disable TLS Versions on vCenter Server Systems. ■ Minor update to Sessions Privileges.
27 APR 2023	■ Minor update to Replace a Default Certificate Using HTTPS PUT. ■ Minor update to Managing ESXi Audit Records.
21 MAR 2023	■ Minor update to Virtual Disk Encryption. ■ Minor update to Encrypt an Existing Virtual Machine or Virtual Disk. ■ Minor update to vSphere Native Key Provider Overview. ■ Minor update to Back up a vSphere Native Key Provider.
07 MAR 2023	■ Minor update to Troubleshooting and Recovering the Secure ESXi Configuration. ■ Minor update to Recover the Secure ESXi Configuration.
15 FEB 2023	■ Minor update to Recover the Secure ESXi Configuration. ■ Minor update to Virtual Machine Encryption Interoperability.

Revision	Description
31 JAN 2023	■ Updated How vSphere Virtual Machine Encryption Protects Your Environment and Prerequisites and Required Privileges for Encryption Tasks to state that ESXi Shell users also have cryptographic operation privileges. ■ Minor update to Enable Virtualization-based Security on a Virtual Machine and Enable Virtualization-based Security on an Existing Virtual Machine.
23 JAN 2023	■ Minor update to vSphere Native Key Provider Overview. ■ Updated Enable Virtual Trusted Platform Module for an Existing Virtual Machine to state that the **Virtual machine.Configuration.Add or remove device** privilege is required.
23 DEC 2022	■ Minor update to Collect a vm-support Package for an ESXi Host That Uses Encryption. ■ Minor update to Enable and Disable Key Persistence on an ESXi Host. ■ Minor update to Remove Virtual Trusted Platform Module from a Virtual Machine.
23 NOV 2022	■ Minor update to vCenter Server System Roles. ■ Updated Key Persistence Overview and Enable and Disable Key Persistence on an ESXi Host with additional information about vSphere Native Key Provider. ■ Minor update to Secure VLANs.
13 OCT 2022	■ Minor update to Managing ESXi Audit Records. ■ Fixed typo in Update a vSphere Native Key Provider. ■ Minor updates to Virtualization-based Security Best Practices, Enable Virtualization-based Security on a Virtual Machine, and Enable Virtualization-based Security on an Existing Virtual Machine. ■ Removed references to the `vifs` command. See the VMware knowledge base article at https://kb.vmware.com/article/78473.
22 AUG 2022	■ Minor update to Renew or Refresh ESXi Certificates. ■ Minor update to Delete a vSphere Native Key Provider. ■ Fixed the example in Create the Key Provider on the Trust Authority Cluster. ■ Rewrote Disable Host Encryption Mode Using the API to use the vCenter Server Managed Object Browser (MOB). ■ Minor update to multiple VMware vSphere Lifecycle Manager privileges topics.
28 JUL 2022	■ Minor update to Requirements for ESXi Certificate Signing Requests. ■ Minor update to Virtual Trusted Platform Module Overview. ■ Minor update to MAC Address Changes. ■ Updated multiple topics to note that you should assign privileges that use VMware vSphere Lifecycle Manager APIs that accept URLs only to administrators or trusted users.
12 JUL 2022	■ Minor update to Change User Validation Settings. ■ Minor update to Renew or Refresh ESXi Certificates. ■ Minor update to Enable Fault Tolerance Encryption. ■ Fixed problem with the formatting of an ESXCLI command in Enable or Disable the execInstalledOnly Enforcement for a Secure ESXi Configuration. ■ Minor update to Disable Unused Display Features. ■ Removed parameters that are now set to TRUE from Disable Unexposed Features. ■ Corrected the steps in Create an Encryption Storage Policy.

Revision	Description
15 JUN 2022	■ Added information about vCenter Server and encrypted communication to Securing vCenter Server Systems and Associated Services. ■ Added information about assigning permissions on vSphere Distributed Switches to Hierarchical Inheritance of Permissions. ■ Minor update to UEFI Secure Boot for ESXi Hosts. ■ Added information about encryption considerations to Virtual Machine Encryption Best Practices. ■ Minor update to vSphere Native Key Provider Overview. ■ Added information about `HostCryptoState` to Disable Host Encryption Mode Using the API. ■ Added a required privilege, **Cryptographic operations.Migrate**, to Create a Virtual Machine with a Virtual Trusted Platform Module and Enable Virtual Trusted Platform Module for an Existing Virtual Machine. This privilege enables a virtual machine to be powered on when DRS starts it on another host. ■ Minor update to Connecting to the Virtual Machine Console Through a Firewall. ■ Minor update to Enable or Disable TLS Versions on ESXi Hosts. ■ Added information about setting permissions to Content Library Privileges. ■ Fixed typo in vSphere Tagging Privileges.
29 APR 2022	■ In Managing ESXi Audit Records, the `viewAudit` program replaces the `auditLogReader` program as of vSphere 7.0 Update 3d. ■ Minor updates to Enable or Disable the Secure Boot Enforcement for a Secure ESXi Configuration and Enable or Disable the execInstalledOnly Enforcement for a Secure ESXi Configuration. ■ Minor updates to How vSphere Virtual Machine Encryption Protects Your Environment. ■ Updated information about vSphere Native Key Provider and key persistence in Key Persistence Overview, Back up a vSphere Native Key Provider, and Enable and Disable Key Persistence on an ESXi Host. ■ Minor update to vSphere Native Key Provider Overview. ■ Fixed commands in Update a vSphere Native Key Provider. ■ Minor update to Connecting ESXi Hosts Through Firewalls. ■ Minor update to Storage Views Privileges.
10 MAR 2022	■ Minor update to Host Upgrades and Certificates. ■ Fixed incorrect commands in step 4 in Use Custom Certificates with Auto Deploy. ■ Updated the prerequisites in Add AMD Secure Encrypted Virtualization-Encrypted State to a Virtual Machine with the vSphere Client, Add AMD Secure Encrypted Virtualization-Encrypted State to a Virtual Machine, Enable AMD Secure Encrypted Virtualization-Encrypted State on an Existing Virtual Machine with the vSphere Client, and Enable AMD Secure Encrypted Virtualization-Encrypted State on an Existing Virtual Machine. ■ Minor update to Virtual Machine Encryption Interoperability. ■ Updated vSphere Native Key Provider Overview to state that the vSphere Native Key Provider does not require a TPM 2.0. ■ In Configure the Trusted Key Provider for Trusted Hosts Using the vSphere Client and Configure the Trusted Key Provider for Trusted Hosts Using the Command Line, clarified that when you add a vSphere Trust Authority key provider, it takes some time for the key provider to become available for use. ■ Added required privileges to Create a Virtual Machine with a Virtual Trusted Platform Module, Enable Virtual Trusted Platform Module for an Existing Virtual Machine, and Remove Virtual Trusted Platform Module from a Virtual Machine.

Revision	Description
19 JAN 2022	■ Minor update to Understanding Authorization in vSphere. ■ Minor update to Restore ESXi Certificate and Key Files. ■ For a standalone ESXi host, clarified that you must run the `reconfigureEsx ESXiHost` command from a vCenter Server system in Enable or Disable TLS Versions on ESXi Hosts. ■ Updated Considerations When Using FIPS with information about the vCenter Server file-based backup and restore.
21 DEC 2021	■ Fixed a typo in Upload an SSH Key Using a vifs Command. ■ Minor update to TPM Sealing Policies Overview. ■ Minor update to Enable or Disable the Secure Boot Enforcement for a Secure ESXi Configuration. ■ Minor update to Resolve ESXi Host Encryption Mode Issues. ■ Minor update to Encrypted vSphere vMotion.
07 DEC 2021	■ Minor update to Comparison of vSphere Key Providers. ■ Added new topic, Import a vSphere Native Key Provider in an Enhanced Linked Mode Configuration. ■ Minor update to vSphere Trust Authority Best Practices, Caveats, and Interoperability. ■ Added new topic, Rekey an Encrypted Virtual Machine Using the vSphere Client. ■ Updated Content Library Privileges with new privileges. ■ Updated vSphere with Tanzu Privileges with new privileges.
03 NOV 2021	■ Minor update to Securing the Virtual Networking Layer. ■ Minor update to Enable or Disable the execInstalledOnly Enforcement for a Secure ESXi Configuration. ■ Updated View Certificate Expiration Information for Multiple ESXi Hosts, Identify Virtual Trusted Platform Module Enabled Virtual Machines, View Virtual Trusted Platform Module Device Certificates, and Identify VBS-Enabled Virtual Machines to reflect a minor UI change with how to show and hide columns. ■ Updated Manage ESXi Firewall Settings and Add Allowed IP Addresses for an ESXi Host to reflect minor UI changes. ■ Added encryption-related information to Virtual Machine Encryption Interoperability. ■ Updated MAC Address Changes and Forged Transmits to reflect the change in the default for both options from Accept to Reject. ■ Corrected a privilege in Virtual Machine Provisioning Privileges. ■ Minor update to Single Sign-On Audit Events.
05 OCT 2021	Initial release.

Security in the vSphere Environment

1

The components of a vSphere environment are secured out of the box by several features such as authentication, authorization, a firewall on each ESXi host, and so on. You can modify the default setup in many ways. For example, you can set permissions on vCenter objects, open firewall ports, or change the default certificates. You can take security measures for different objects in the vCenter object hierarchy, for example, vCenter Server systems, ESXi hosts, virtual machines, and network and storage objects.

A high-level overview of different areas of vSphere that require attention helps you plan your security strategy. You also benefit from other vSphere Security resources on the VMware Web site.

Read the following topics next:

- Securing the ESXi Hypervisor
- Securing vCenter Server Systems and Associated Services
- Securing Virtual Machines
- Securing the Virtual Networking Layer
- Passwords in Your vSphere Environment
- Security Best Practices and Resources

Securing the ESXi Hypervisor

The ESXi hypervisor is secured out of the box. You can further protect ESXi hosts by using lockdown mode and other built-in features. For consistency, you can set up a reference host and keep all hosts in sync with the host profile of the reference host. You can also protect your environment by performing scripted management, which ensures that changes apply to all hosts.

You can enhance protection of ESXi hosts that are managed by vCenter Server with the following actions. See the *Security of the VMware vSphere Hypervisor* white paper for background and details.

Limit ESXi access

By default, the ESXi Shell and SSH services are not running and only the root user can log in to the Direct Console User Interface (DCUI). If you decide to enable ESXi or SSH access, you can set timeouts to limit the risk of unauthorized access.

Users who can access the ESXi host must have permissions to manage the host. You set permissions on the host object from the vCenter Server system that manages the host.

Use named users and least privilege

By default, the root user can perform many tasks. Do not allow administrators to log in to the ESXi host using the root user account. Instead, create named administrator users from vCenter Server and assign those users the Administrator role. You can also assign those users a custom role. See Create a vCenter Server Custom Role.

If you manage users directly on the host, role management options are limited. See the *vSphere Single Host Management - VMware Host Client* documentation.

Minimize the number of open ESXi firewall ports

By default, firewall ports on your ESXi host are opened only when you start a corresponding service. You can use the vSphere Client or ESXCLI or PowerCLI commands to check and manage firewall port status.

See ESXi Firewall Configuration.

Automate ESXi host management

Because it is often important that different hosts in the same data center are in sync, use scripted installation or vSphere Auto Deploy to provision hosts. You can manage the hosts using scripts. Host profiles are an alternative to scripted management. You set up a reference host, export the host profile, and apply the host profile to all hosts. You can apply the host profile directly or as part of provisioning with Auto Deploy.

See Use Scripts to Manage Host Configuration Settings and see the *vCenter Server Installation and Setup* documentation for information about vSphere Auto Deploy.

Take advantage of lockdown mode

In lockdown mode, ESXi hosts can be accessed only through vCenter Server by default. You can select strict lockdown mode or normal lockdown mode. You can define Exception Users to allow direct access to service accounts such as backup agents.

See Lockdown Mode.

Check VIB package integrity

Each VIB package has an associated acceptance level. You can add a VIB to an ESXi host only if the VIB acceptance level is the same or better than the acceptance level of the host. You cannot add a CommunitySupported or PartnerSupported VIB to a host unless you explicitly change the host's acceptance level.

See Manage the Acceptance Levels of Hosts and VIBs.

Manage ESXi certificates

The VMware Certificate Authority (VMCA) provisions each ESXi host with a signed certificate that has VMCA as the root certificate authority by default. If your company policy requires it, you can replace the existing certificates with certificates that are signed by a third-party or an enterprise CA.

See Certificate Management for ESXi Hosts.

Consider Smart card authentication

ESXi supports the use of smart card authentication instead of user name and password authentication. For additional security, you can configure smart card authentication. Two-factor authentication is also supported for vCenter Server. You can configure user name and password authentication and smart card authentication at the same time.

See Configuring Smart Card Authentication for ESXi.

Consider ESXi account lockout

Account locking is supported for access through SSH and through the vSphere Web Services SDK. By default, a maximum of 10 failed attempts is allowed before the account is locked. The account is unlocked after two minutes by default.

Note The Direct Console Interface (DCUI) and the ESXi Shell do not support account lockout.

See ESXi Passwords and Account Lockout.

Security considerations for standalone hosts are similar, though the management tasks might differ. See the *vSphere Single Host Management - VMware Host Client* documentation.

Securing vCenter Server Systems and Associated Services

Your vCenter Server system and associated services are protected by authentication through vCenter Single Sign-On and by authorization through the vCenter Server permissions model. You can modify the default behavior, and you can take steps to limit access to your environment.

As you protect your vSphere environment, consider that all services that are associated with the vCenter Server instances must be protected. In some environments, you might protect several vCenter Server instances.

vCenter and encrypted communication

By default ("out of the box"), all data communication between vCenter Server and other vSphere components is encrypted. In some cases, depending on how you configure your environment, some traffic might be unencrypted. For example, you can configure unencrypted SMTP for email alerts and unencrypted SNMP for monitoring. DNS traffic is also unencrypted. vCenter Server listens on port 80 (TCP) and port 443 (TCP). Port 443 (TCP) is the industry-standard HTTPS (secure HTTP) port and uses TLS 1.2 encryption for protection.

Port 80 (TCP) is the industry-standard HTTP port and does not use encryption. The purpose of port 80 is to redirect requests from port 80 to port 443, where they are secure.

Harden all vCenter host machines

The first step in protecting your vCenter environment is hardening each machine on which vCenter Server or an associated service runs. Similar considerations apply to a physical machine or a virtual machine. Always install the latest security patches for your operating system and follow industry standard best practices to protect the host machine.

Learn about the vCenter certificate model

By default, the VMware Certificate Authority provisions each ESXi host and each machine in the environment with a certificate signed by VMCA. If your company policy requires it, you can change the default behavior. See the *vSphere Authentication* documentation for details.

For additional protection, explicitly remove expired or revoked certificates and failed installations.

Configure vCenter Single Sign-On

vCenter Server and associated services are protected by the vCenter Single Sign-On authentication framework. When you first install the software, you specify a password for the administrator of the vCenter Single Sign-On domain, administrator@vsphere.local by default. Only that domain is initially available as an identity source. You can add an external identity provider, such as Microsoft Active Directory Federation Services (AD FS), for federated authentication. You can add other identity sources, either Active Directory or LDAP, and set a default identity source. Users who can authenticate to one of those identity sources can view objects and perform tasks if they are authorized to do so. See the *vSphere Authentication* documentation for details.

Assign roles to named users or groups

For better logging, associate each permission that you give on an object with a named user or group and a predefined role or custom role. The vSphere permissions model allows great flexibility through multiple ways of authorizing users or groups. See Understanding Authorization in vSphere and Required Privileges for Common Tasks.

Restrict administrator privileges and the use of the administrator role. If possible, do not use the anonymous Administrator user.

Set up PTP or NTP

Set up PTP or NTP for each node in your environment. The certificate infrastructure requires an accurate time stamp and does not work correctly if the nodes are out of sync.

See Synchronizing Clocks on the vSphere Network.

Securing Virtual Machines

To secure your virtual machines, keep the guest operating systems patched and protect your environment just as you protect your physical machine. Consider disabling unnecessary functionality, minimize the use of the virtual machine console, and follow other best practices.

Protect the guest operating system

To protect your guest operating system, make sure that it uses the most recent patches and, if appropriate, anti-spyware and anti-malware applications. See the documentation from your guest operating system vendor and, potentially, other information available in books or on the Internet for that operating system.

Disable unnecessary functionality

Check that unnecessary functionality is disabled to minimize potential points of attack. Many of the features that are used infrequently are disabled by default. Remove unnecessary hardware and disable certain features such as host-guest filesystem (HGFS) or copy and paste between the virtual machine and a remote console.

See Disable Unnecessary Functions Inside Virtual Machines.

Use templates and scripted management

Virtual machine templates enable you to set up the operating system so that it meets your requirements, and to create other VMs with the same settings.

If you want to change virtual machine settings after initial deployment, consider using scripts, for example, PowerCLI. This documentation explains how to perform tasks using the GUI. Consider using scripts instead of the GUI to keep your environment consistent. In large environments, you can group virtual machines into folders to optimize scripting.

For information on templates, see Use Templates to Deploy Virtual Machines and the *vSphere Virtual Machine Administration* documentation. For information on PowerCLI, see the VMware PowerCLI documentation.

Minimize use of the virtual machine console

The virtual machine console provides the same function for a virtual machine that a monitor on a physical server provides. Users with access to a virtual machine console have access to virtual machine power management and to removable device connectivity controls. As a result, virtual machine console access might allow a malicious attack on a virtual machine.

Consider UEFI secure boot

You can configure your virtual machine to use UEFI boot. If the operating system supports secure UEFI boot, you can select that option for your VMs for additional security. See Enable or Disable UEFI Secure Boot for a Virtual Machine.

Consider Carbon Black Cloud Workload

You can install and use Carbon Black Cloud Workload to identify risk, prevent attacks, and detect unusual activities. With the AppDefense functionality built in to the Carbon Black Cloud platform, Carbon Black Cloud Workload is the successor product for AppDefense.

Securing the Virtual Networking Layer

The virtual networking layer includes virtual network adapters, virtual switches, distributed virtual switches, and ports and port groups. ESXi relies on the virtual networking layer to support communications between VMs and their users. In addition, ESXi uses the virtual networking layer to communicate with iSCSI SANs, NAS storage, and so on.

vSphere includes the full array of features necessary for a secure networking infrastructure. You can secure each element of the infrastructure, such as virtual switches, distributed virtual switches, and virtual network adapters, separately. In addition, consider the following guidelines, discussed in more detail in Chapter 13 Securing vSphere Networking.

Isolate network traffic

Isolation of network traffic is essential to a secure ESXi environment. Different networks require different access and level of isolation. A management network isolates client traffic, command-line interface (CLI) or API traffic, and third-party software traffic from normal traffic. Ensure that the management network is accessible only by system, network, and security administrators.

See ESXi Networking Security Recommendations.

Use firewalls to secure virtual network elements

You can open and close firewall ports and secure each element in the virtual network separately. For ESXi hosts, firewall rules associate services with corresponding firewalls and can open and close the firewall according to the status of the service.

You can also open ports on vCenter Server instances explicitly.

For the list of all supported ports and protocols in VMware products, including vSphere and vSAN, see the VMware Ports and Protocols Tool™ at https://ports.vmware.com/. You can search ports by VMware product, create a customized list of ports, and print or save port lists.

Consider network security policies

Network security policies provide protection of traffic against MAC address impersonation and unwanted port scanning. The security policy of a standard or distributed switch is implemented in Layer 2 (Data Link Layer) of the network protocol stack. The three elements of the security policy are promiscuous mode, MAC address changes, and forged transmits.

See the *vSphere Networking* documentation for instructions.

Secure VM networking

The methods that you use to secure VM networking depend on several factors, including:

- The guest operating system that is installed
- Whether the VMs operate in a trusted environment

Virtual switches and distributed virtual switches provide significant protection when used with other common security practices, such as installing firewalls.

See Chapter 13 Securing vSphere Networking.

Consider VLANs to protect your environment

ESXi supports IEEE 802.1q VLANs. VLANs let you segment a physical network. You can use VLANs to further protect the VM network or storage configuration. When you use VLANS, two VMs on the same physical network cannot send packets to or receive packets from each other unless they are on the same VLAN.

See Securing Virtual Machines with VLANs.

Secure connections to virtualized storage

A VM stores operating system files, application files, and other data on a virtual disk. Each virtual disk appears to the VM as a SCSI drive that is connected to a SCSI controller. A VM is isolated from storage details and cannot access the information about the LUN where its virtual disk resides.

The Virtual Machine File System (VMFS) is a distributed file system and volume manager that presents virtual volumes to the ESXi host. You are responsible for securing the connection to storage. For example, if you are using iSCSI storage, you can set up your environment to use CHAP. If required by company policy, you can set up mutual CHAP. Use the vSphere Client or CLIs to set up CHAP.

See Storage Security Best Practices.

Evaluate the use of IPSec

ESXi supports IPSec over IPv6. You cannot use IPSec over IPv4.

See Internet Protocol Security.

Passwords in Your vSphere Environment

Password restrictions, password expiration, and account lockout in your vSphere environment depend on the system that the user targets, who the user is, and how policies are set.

ESXi Passwords

ESXi password restrictions are determined by certain requirements. See ESXi Passwords and Account Lockout.

Passwords for vCenter Server and Other vCenter Services

vCenter Single Sign-On manages authentication for all users who log in to vCenter Server and other vCenter services. The password restrictions, password expiration, and account lockout depend on the user's domain and on who the user is.

vCenter Single Sign-On Administrator

The password for the administrator@vsphere.local user, or the administrator@*mydomain* user if you selected a different domain during installation, does not expire and is not subject to the lockout policy. In all other regards, the password must follow the restrictions that are set in the vCenter Single Sign-On password policy. See *vSphere Authentication* for details.

If you forget the password for this user, search the VMware Knowledge Base system for information on resetting this password. The reset requires additional privileges such as root access to the vCenter Server system.

Other Users of the vCenter Single Sign-On Domain

Passwords for other vsphere.local users, or users of the domain that you specified during installation, must follow the restrictions that are set by the vCenter Single Sign-On password policy and lockout policy. See *vSphere Authentication* for details. These passwords expire after 90 days by default. Administrators can change the expiration as part of the password policy.

If you forget your vsphere.local password, an administrator user can reset the password using the `dir-cli` command.

Other Users

Password restrictions, password expiration, and account lockout for all other users are determined by the domain (identity source) to which the user can authenticate.

vCenter Single Sign-On supports one default identity source. Users can log in to the corresponding domain with the vSphere Client with their user names. If users want to log in to a non-default domain, they can include the domain name, that is, specify *user*@*domain* or *domain**user*. The domain password parameters apply to each domain.

Passwords for vCenter Server Direct Console User Interface Users

The vCenter Server appliance is a preconfigured virtual machine that is optimized for running vCenter Server and the associated services.

When you deploy the vCenter Server, you specify these passwords.

- Password for the root user.
- Password for the administrator of the vCenter Single Sign-On domain, administrator@vsphere.local by default.

You can change the root user password and perform other vCenter Server local user management tasks from the vCenter Server Management Interface. See *vCenter Server Configuration*.

Security Best Practices and Resources

If you follow best practices, your ESXi and vCenter Server can be as secure as or even more secure than an environment that does not include virtualization.

This manual includes best practices for the different components of your vSphere infrastructure.

Table 1-1. Security Best Practices

vSphere component	Resource
ESXi host	Chapter 3 Securing ESXi Hosts
vCenter Server system	vCenter Server Security Best Practices
Virtual machine	Virtual Machine Security Best Practices
vSphere Networking	vSphere Networking Security Best Practices

This manual is only one of the sources you must use to ensure a secure environment.

VMware security resources, including security alerts and downloads, are available on the Web.

Table 1-2. VMware Security Resources on the Web

Topic	Resource
Information on ESXi and vCenter Server security and operations, including secure configuration and hypervisor security.	https://core.vmware.com/security
VMware security policy, up-to-date security alerts, security downloads, and focus discussions of security topics.	http://www.vmware.com/go/security
Corporate security response policy	http://www.vmware.com/support/policies/security_response.html VMware is committed to helping you maintain a secure environment. Security issues are corrected in a timely manner. The VMware Security Response Policy states our commitment to resolve possible vulnerabilities in our products.

Table 1-2. VMware Security Resources on the Web (continued)

Topic	Resource
Third-party software support policy	http://www.vmware.com/support/policies/ VMware supports a variety of storage systems, software agents such as backup agents, system management agents, and so forth. You can find lists of agents, tools, and other software that supports ESXi by searching http://www.vmware.com/vmtn/resources/ for ESXi compatibility guides. The industry offers more products and configurations than VMware can test. If VMware does not list a product or configuration in a compatibility guide, Technical Support attempts to help you with any problems, but cannot guarantee that the product or configuration can be used. Always evaluate security risks for unsupported products or configurations carefully.
Compliance and security standards, and partner solutions and in-depth content about virtualization and compliance	https://core.vmware.com/compliance
Information on security certifications and validations such as CCEVS and FIPS for different versions of the components of vSphere.	https://www.vmware.com/support/support-resources/certifications.html
Security configuration guides (formerly known as hardening guides) for different versions of vSphere and other VMware products.	https://core.vmware.com/security
Security of the VMware vSphere Hypervisor white paper	http://www.vmware.com/files/pdf/techpaper/vmw-wp-secrty-vsphr-hyprvsr-uslet-101.pdf

vSphere Permissions and User Management Tasks

2

Authentication and authorization govern access. vCenter Single Sign-On supports authentication, which means it determines whether a user can log in to vSphere components at all. Each user must also be authorized to view or manipulate vSphere objects.

vSphere supports several different authorization mechanisms, discussed in Understanding Authorization in vSphere. This section focuses on how the vCenter Server permission model works and how to perform user management tasks.

vCenter Server allows fine-grained control over authorization with permissions and roles. When you assign a permission to an object in the vCenter Server object hierarchy, you specify which user or group has which privileges on that object. To specify the privileges, you use roles, which are sets of privileges.

Initially, only the administrator user for the vCenter Single Sign-On domain is authorized to log in to the vCenter Server system. The default domain is vsphere.local and the default administrator is administrator@vsphere.local. You can change the default domain during installation of vSphere.

The administrator user can proceed as follows:

1. Add an identity source in which users and groups are defined to vCenter Single Sign-On. See the *vSphere Authentication* documentation.
2. Give privileges to a user or group by selecting an object such as a virtual machine or a vCenter Server system and assigning a role on that object for the user or group.

 (Assigning Roles and Permissions Using the vSphere Client)

Read the following topics next:

- Understanding Authorization in vSphere
- Managing Permissions for vCenter Components
- Global Permissions
- Using Roles to Assign Privileges
- Best Practices for Roles and Permissions
- Required Privileges for Common Tasks

Understanding Authorization in vSphere

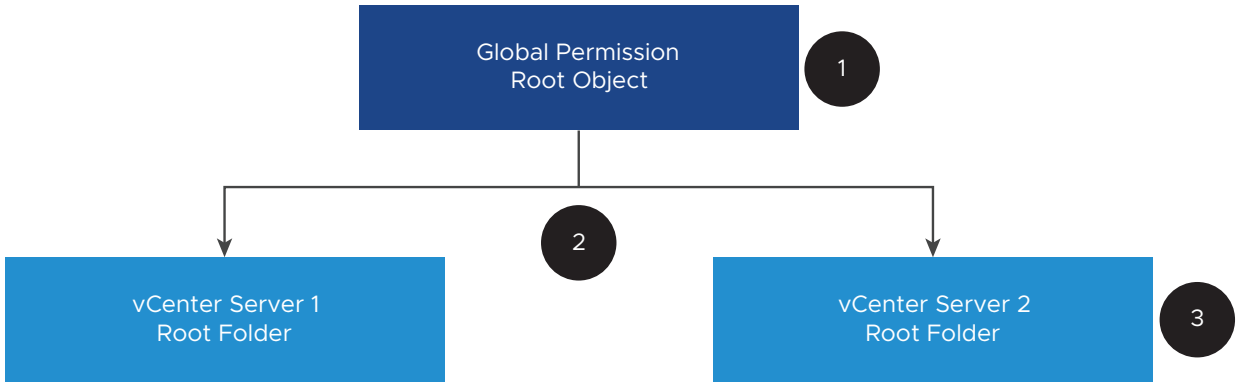
vSphere supports several models for determining whether a user is allowed to perform a task. Group membership in a vCenter Single Sign-On group decides what you are allowed to do. Your role on an object or your global permission determines whether you are allowed to perform other tasks.

Authorization Overview

vSphere allows privileged users to give other users permissions to perform tasks. You can use global permissions, or you can use local vCenter Server permissions to authorize other users for individual vCenter Server instances.

The following figure illustrates how global and local permissions work.

Figure 2-1. Global Permissions and Local Permissions



In this figure:

1 You assign a global permission at the root object level with "Propagate to children" selected.

2 vCenter Server propagates the permissions to the vCenter Server 1 and vCenter Server 2 object hierarchies in the environment.

3 A local permission on the root folder on vCenter Server 2 overrides the global permission.

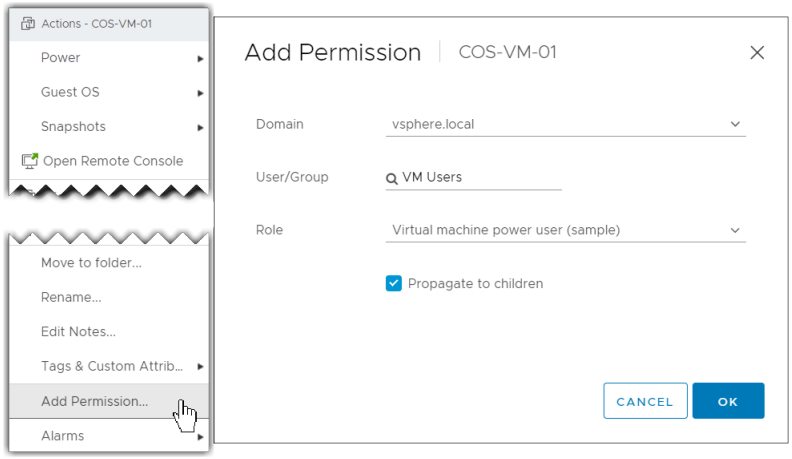
vCenter Server Permissions

The permission model for vCenter Server systems relies on assigning permissions to objects in the object hierarchy. Users get permissions in the following ways.

- From a specific permission for the user or from the groups that the user is a member of
- From a permission on the object or through the permission inheritance from a parent object

Each permission gives one user or group a set of privileges, that is, a role for a selected object. You can use the vSphere Client to add permissions. For example, you can right-click a virtual machine, select **Add Permission**, and complete the dialog box to assign a role to a group of users. That role gives those users the corresponding privileges on the virtual machine.

Figure 2-2. Adding Permissions to a Virtual Machine Using the vSphere Client



Global Permissions

Global permissions give a user or group privileges to view or manage all objects in each of the inventory hierarchies of the solutions in the deployment. That is, global permissions are applied to a global root object that spans solution inventory hierarchies. (Solutions include vCenter Server, vRealize Orchestrator, and so on.) Global permissions also apply to global objects such as tags and content libraries. For example, consider a deployment that consists of two solutions, vCenter Server and vRealize Orchestrator. You can use global permissions to assign a role to a group of users that has read-only privileges to all objects in both the vCenter Server and vRealize Orchestrator object hierarchies.

Global permissions are replicated across the vCenter Single Sign-On domain (vsphere.local by default). Global permissions do not provide authorization for services managed through the vCenter Single Sign-On domain groups. See Global Permissions.

Group Membership in vCenter Single Sign-On Groups

Members of a vCenter Single Sign-On domain group can perform certain tasks. For example, you can perform license management if you are a member of the LicenseService.Administrators group. See the *vSphere Authentication* documentation.

ESXi Local Host Permissions

If you are managing a standalone ESXi host that is not managed by a vCenter Server system, you can assign one of the predefined roles to users. See the *vSphere Single Host Management - VMware Host Client* documentation.

For managed hosts, assign roles to the ESXi host object in the vCenter Server inventory.

Understanding the Object-Level Permission Model

You authorize a user or group to perform tasks on vCenter Server objects by using permissions on the object. From a programmatic standpoint, when a user tries to perform an operation, an API method is executed. vCenter Server checks the permissions for that method to see if the user is authorized to perform the operation. For example, when a user tries to add a host, the `AddStandaloneHost_Task` method is invoked. This method requires that the role for the user has the **Host.Inventory.Add standalone host** privilege. If the check does not find this privilege, the user is denied permission to add the host.

The following concepts are important.

Permissions

Each object in the vCenter Server object hierarchy has associated permissions. Each permission specifies for one group or user which privileges that group or user has on the object. Permissions can propagate to child objects.

Users and Groups

On vCenter Server systems, you can assign privileges only to authenticated users or groups of authenticated users. Users are authenticated through vCenter Single Sign-On. Users and groups must be defined in the identity source that vCenter Single Sign-On uses to authenticate. Define users and groups using the tools in your identity source, for example, Active Directory.

Privileges

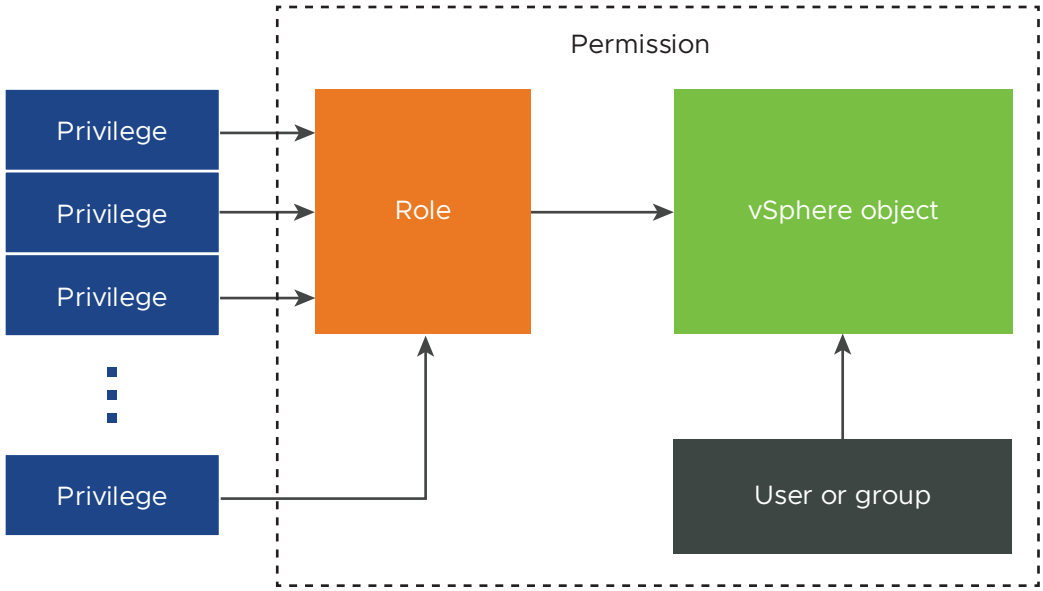
Privileges are fine-grained access controls. You can group those privileges into roles, which you can then map to users or groups.

Roles

Roles are sets of privileges. Roles allow you to assign permissions on an object based on a typical set of tasks that users perform. System roles, such as Administrator, are predefined on vCenter Server and cannot be changed. vCenter Server also provides some default sample roles, such as Resource Pool Administrator, that you can modify. You can create custom roles either from scratch or by cloning and modifying sample roles. See Create a vCenter Server Custom Role.

The following figure illustrates how a permission is constructed from privileges and roles, and assigned to a user or group for a vSphere object.

Figure 2-3. vSphere Permissions



To assign permissions to an object, you follow these steps:

1 Select the object to which you want to apply the permission in the vCenter Server object hierarchy.

2 Select the group or user that should have privileges on the object.

3 Select individual privileges or a role, that is a set of privileges, that the group or user should have on the object.

   By default, Propagate to children is not selected. You must select the checkbox for the group or user to have the selected role on the selected object and its child objects.

vCenter Server offers sample roles, which combine frequently used privilege sets. You can also create custom roles by combining a set of roles.

Permissions must often be defined on both a source object and a destination object. For example, if you move a virtual machine, you need privileges on that virtual machine, but also privileges on the destination data center.

See the following information.

To find out about...	See...
Creating custom roles.	Create a vCenter Server Custom Role
All privileges and the objects to which you can apply the privileges	Chapter 16 Defined Privileges
Sets of privileges that are required on different objects for different tasks.	Required Privileges for Common Tasks

The permissions model for standalone ESXi hosts is simpler. See Assigning Privileges for ESXi Hosts.

vCenter Server User Validation

vCenter Server systems that use a directory service regularly validate users and groups against the user directory domain. Validation occurs at regular intervals specified in the vCenter Server settings. For example, assume that user Smith is assigned a role on several objects. The domain administrator changes the name to Smith2. The host concludes that Smith no longer exists and removes permissions associated with that user from the vSphere objects when the next validation occurs.

Similarly, if user Smith is removed from the domain, all permissions associated with that user are removed when the next validation occurs. If a new user Smith is added to the domain before the next validation occurs, the new user Smith replaces the old user Smith in permissions on any object.

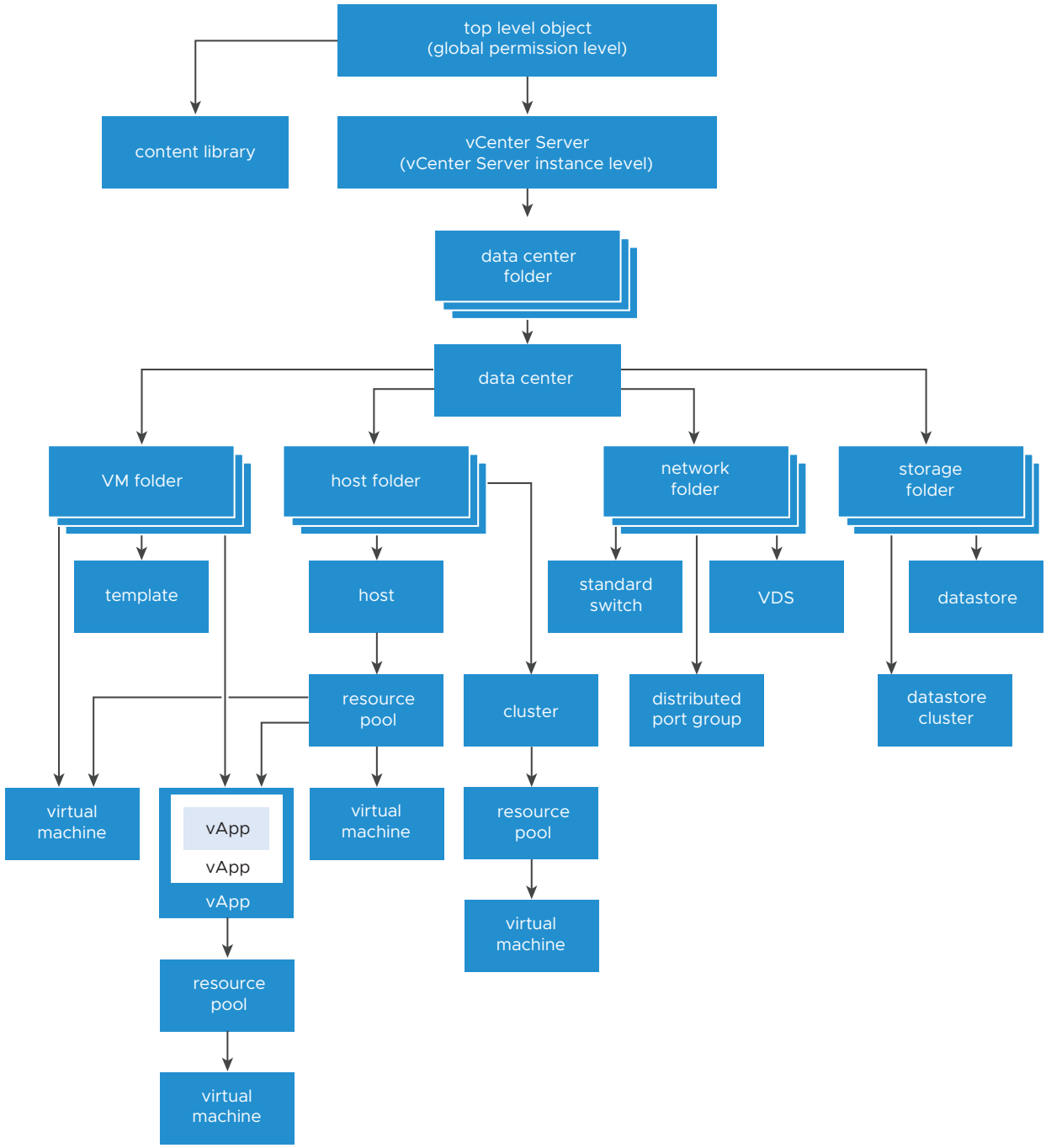
Hierarchical Inheritance of Permissions

When you assign a permission to an object, you can choose whether the permission propagates down the object hierarchy. You set propagation for each permission. Propagation is not universally applied. Permissions defined for a child object always override the permissions that are propagated from parent objects.

The following figure illustrates the inventory hierarchy and the paths by which permissions can propagate.

Note Global permissions support assigning privileges across solutions from a global root object. See Global Permissions.

Figure 2-4. vSphere Inventory Hierarchy



About this figure:

- You cannot set direct permissions on the VM, host, network, and storage folders. That is, these folders act as containers, and as such are not visible to users.
- You cannot set permissions on standard switches.

Note To be able to set and propagate permissions to children on a vSphere Distributed Switch (VDS), the switch object must reside in a network folder created on the data center.

Most inventory objects inherit permissions from a single parent object in the hierarchy. For example, a datastore inherits permissions from either its parent datastore folder or parent data center. Virtual machines inherit permissions from both the parent virtual machine folder and the parent host, cluster, or resource pool simultaneously.

For example, you can set permissions for a distributed switch and its associated distributed port groups, by setting permissions on a parent object, such as a folder or data center. You must also select the option to propagate these permissions to child objects.

Permissions take several forms in the hierarchy:

Managed entities

Managed entities refer to the following vSphere objects. Managed entities offer specific operations that vary depending on the entity type. Privileged users can define permissions on managed entities. See the vSphere API documentation for more information about vSphere objects, properties, and methods.

- Clusters
- Data centers
- Datastores
- Datastore clusters
- Folders
- Hosts
- Networks (except vSphere Distributed Switches)
- Distributed port groups
- Resource pools
- Templates
- Virtual machines
- vSphere vApps

Global entities

You cannot modify permissions on entities that derive permissions from the root vCenter Server system.

- Custom fields
- Licenses
- Roles
- Statistics intervals
- Sessions

Multiple Permission Settings

Objects might have multiple permissions, but only one permission for each user or group. For example, one permission might specify that GroupAdmin has the Administrator role on an object. Another permission might specify that the GroupVMAdmin has the Virtual Machine Administrator role on the same object. However, the GroupVMAdmin group cannot have another permission for the same GroupVMAdmin on this object.

A child object inherits the permissions of its parent if the parent's propagate property is set to true. A permission that is set directly on a child object overrides the permission in the parent object. See Example 2: Child Permissions Overriding Parent Permissions.

If multiple group roles are defined on the same object, and a user belongs to two or more of those groups, two situations are possible:

- No permission for the user is defined directly on the object. In that case, the user gets the union of the permissions that the groups have on the object.
- A permission for the user is defined directly on the object. In that case, the permissions for the user take precedence over all group permissions.

Example 1: Permission Inheritance from Multiple Groups

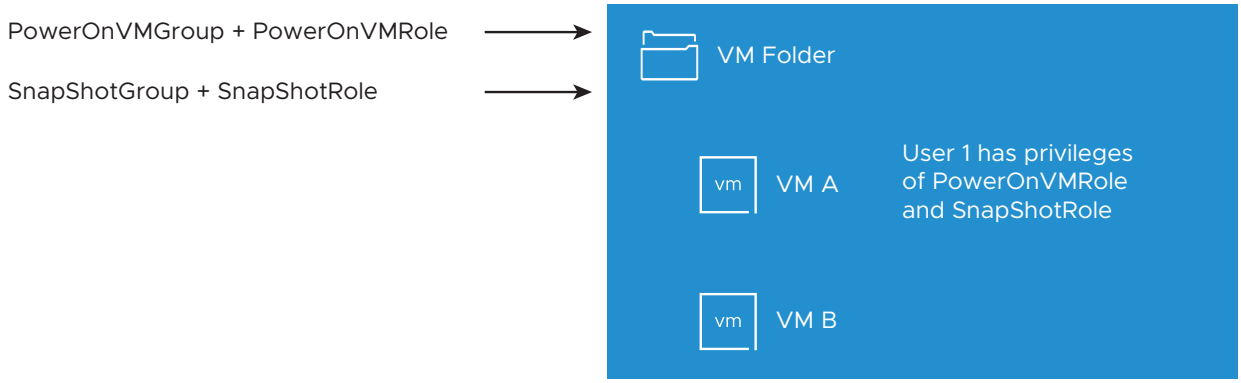
This example illustrates how an object can inherit multiple permissions from groups that are granted permission on a parent object.

In this example, two permissions are assigned on the same object for two different groups.

- PowerOnVMRole can power on virtual machines.
- SnapShotRole can take snapshots of virtual machines.
- PowerOnVMGroup is granted the PowerOnVMRole on VM Folder, with the permission set to propagate to child objects.
- SnapShotGroup is granted the SnapShotRole on VM Folder, with the permission set to propagate to child objects.
- User 1 is not assigned specific privileges.

User 1, who belongs to both the PowerOnVMGroup and the SnapShotGroup, logs in. User 1 can both power on and take snapshots of both VM A and VM B.

Figure 2-5. Example 1: Permission Inheritance from Multiple Groups



Example 2: Child Permissions Overriding Parent Permissions

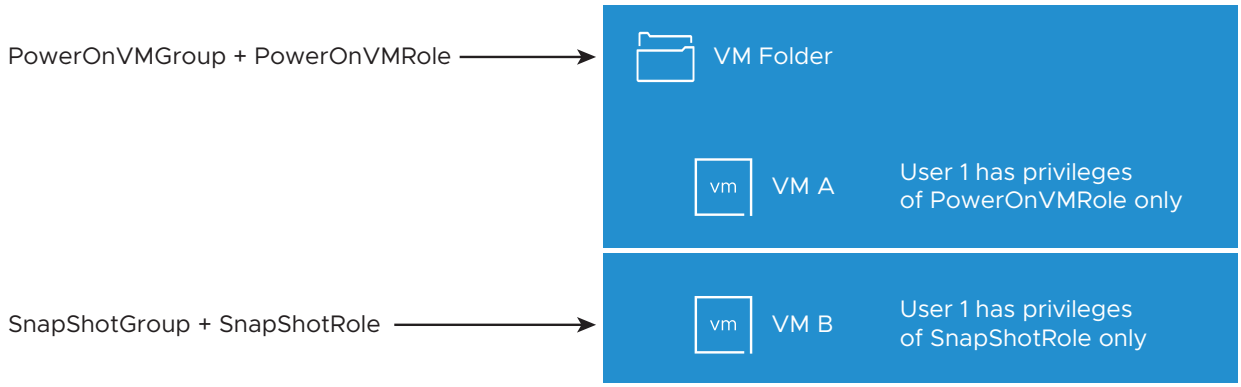
This example illustrates how permissions that are assigned on a child object can override permissions that are assigned on a parent object. You can use this overriding behavior to restrict user access to particular areas of the inventory.

In this example, permissions are defined on two different objects for two different groups.

- PowerOnVMRole can power on virtual machines.
- SnapShotRole can take snapshots of virtual machines.
- PowerOnVMGroup is granted the PowerOnVMRole on VM Folder, with the permission set to propagate to child objects.
- SnapShotGroup is granted the SnapShotRole on VM B.

User 1, who belongs to both the PowerOnVMGroup and the SnapShotGroup, logs in. Because the SnapShotRole is assigned at a lower point in the hierarchy than the PowerOnVMRole, it overrides PowerOnVMRole on VM B. User 1 can power on VM A, but not take snapshots. User 1 can take snapshots of VM B, but not power it on.

Figure 2-6. Example 2: Child Permissions Overriding Parent Permissions



Example 3: User Role Overriding Group Role

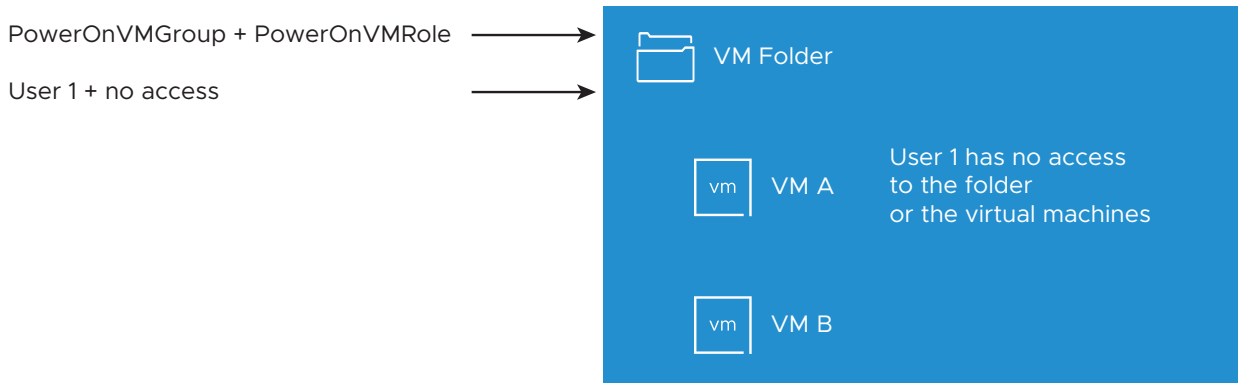
This example illustrates how the role assigned directly to an individual user overrides the privileges associated with a role assigned to a group.

In this example, permissions are defined on the same object. One permission associates a group with a role, the other permission associates an individual user with a role. The user is a member of the group.

- PowerOnVMRole can power on virtual machines.
- PowerOnVMGroup is granted the PowerOnVMRole on VM Folder.
- User 1 is granted the NoAccess role on VM Folder.

User 1, who belongs to PowerOnVMGroup, logs in. The NoAccess role granted to User 1 on VM Folder overrides the role assigned to the group. User 1 has no access to VM Folder or VMs A and B. VMs A and B are not visible in the hierarchy to User 1.

Figure 2-7. Example 3: User Permissions Overriding Group Permissions



Managing Permissions for vCenter Components

A permission is set on an object in the vCenter object hierarchy. Each permission associates the object with a group or user and the group's or user's access role. For example, you can select a virtual machine object, add one permission that gives the ReadOnly role to Group 1, and add a second permission that gives the Administrator role to User 2.

By assigning a different role to a group of users on different objects, you control the tasks that those users can perform in your vSphere environment. For example, to allow a group to configure memory for the host, select that host and add a permission that grants a role to that group that includes the **Host.Configuration.Memory Configuration** privilege.

For conceptual information about permissions, see the discussion in Understanding the Object-Level Permission Model.

You can assign permissions to objects at different levels of the hierarchy, for example, you can assign permissions to a host object or to a folder object that includes all host objects. See Hierarchical Inheritance of Permissions. You can also assign propagating permissions to a global root object to apply the permissions to all object in all solutions. See Global Permissions.

Add a Permission to an Inventory Object

After you create users and groups and define roles, you must assign the users and groups and their roles to the relevant inventory objects. You can assign the same propagating permissions to multiple objects simultaneously by moving the objects into a folder and setting the permissions on the folder.

When you assign permissions, the user and the group names must match Active Directory precisely, including case. If you upgraded from earlier versions of vSphere, check for case inconsistencies if you experience problems with groups.

Prerequisites

On the object whose permissions you want to modify, you must have a role that includes the **Permissions.Modify permission** privilege.

Procedure

1. Browse to the object for which you want to assign permissions in the vSphere Client object navigator.
2. Click the **Permissions** tab.
3. Click **Add**.
4. (Optional) If you have configured an external identity provider for federated authentication, the domain of that identity provider is available to select in the **Domain** drop-down menu.
5. Select the user or group that will have the privileges defined by the selected role.
   a. From the **Domain** drop-down menu, select the domain for the user or group.
   b. Enter a name in the Search box.

      The system searches user names and group names.
   c. Select the user or group.
6. Select a role from the **Role** drop-down menu.
7. (Optional) To propagate the permissions, select the **Propagate to children** check box.

   The role is applied to the selected object and propagates to the child objects.
8. Click **OK**.

Change or Remove Permissions

After a user or group and role pair is set for an inventory object, you can change the role paired with the user or group or change the setting of the **Propagate to children** check box. You can also remove the permission setting.

Procedure

1 Browse to the object in the vSphere Client object navigator.

2 Click the **Permissions** tab.

3 Click a row to select a permission.

Task	Steps
Change permissions	a Click the **Change Role** icon. b Select a role for the user or group from the **Role** drop-down menu. c Toggle the **Propagate to children** check box to change permission inheritance. d Click **OK**.
Remove permissions	Click the **Remove Permission** icon.

Change User Validation Settings

vCenter Server periodically validates its user and group lists against the users and groups in the user directory. It then removes users or groups that no longer exist in the domain. You can disable validation or change the interval between validations. If you have domains with thousands of users or groups, or if searches take a long time to complete, consider adjusting the search settings.

For vCenter Server versions before vCenter Server 5.0, these settings apply to an Active Directory associated with vCenter Server. For vCenter Server 5.0 and later, these settings apply to vCenter Single Sign-On identity sources.

Note This procedure applies only to vCenter Server user lists. You cannot search ESXi user lists in the same way.

Procedure

1 Browse to the vCenter Server system in the vSphere Client object navigator.

2 Select **Configure** and click **Settings > General**.

3 Click **Edit** and select **User directory**.

4 Change the values as needed and click **Save**.

Option	Description
User directory timeout	Timeout interval, in seconds, for connecting to the Active Directory server. This value specifies the maximum amount of time vCenter Server allows a search to run on the selected domain. Searching large domains can take a long time.
Query limit	Toggle on to set a maximum number of users and groups that vCenter Server displays.
Query limit size	Maximum number of users and groups from the selected domain that vCenter Server displays in the **Select Users or Groups** dialog box. If you enter 0 (zero), all users and groups appear.

Global Permissions

Global permissions are applied to a global root object that spans solutions. In an on-premises SDDC, global permissions might span both vCenter Server and vRealize Orchestrator. But for any vSphere SDDC, global permissions apply to global objects such as tags and content libraries.

You can assign global permissions to users or groups, and decide on the role for each user or group. The role determines the set of privileges that the user or group has for all objects in the hierarchy. You can assign a predefined role or create custom roles. See Using Roles to Assign Privileges.

It is important to distinguish between vCenter Server permissions and global permissions.

vCenter Server permissions

You usually apply a permission to a vCenter Server inventory object such as a virtual machine. When you do, you specify that a user or group has a role (set of privileges) on the object.

Global permissions

Global permissions give a user or group privileges to view or manage all objects in each of the inventory hierarchies in your deployment. Global permissions also apply to global objects such as tags and content libraries. See Permissions on Tag Objects.

If you assign a global permission and do not select Propagate, the users or groups associated with this permission do not have access to the objects in the hierarchy. They only have access to some global functionality such as creating roles.

Important Use global permissions with care. Verify that you really want to assign permissions to all objects in all inventory hierarchies.

Add a Global Permission

You can use global permissions to give a user or group privileges for all objects in all inventory hierarchies in your deployment.

Important Use global permissions with care. Verify that you really want to assign permissions to all objects in all inventory hierarchies.

Prerequisites

To perform this task, you must have **Permissions.Modify permission** privileges on the root object for all inventory hierarchies.

Procedure

1. Log in to the vCenter Server by using the vSphere Client.
2. Select **Administration** and click **Global Permissions** in the Access Control area.
3. Select the domain from the **Permissions provider** drop-down menu.
4. (Optional) If you have configured an external identity provider for federated authentication, the domain of that identity provider is available to select in the **Domain** drop-down menu.
5. Click **Add**.
6. Select the user or group that will have the privileges defined by the selected role.
   a. From the **Domain** drop-down menu, select the domain for the user or group.
   b. Enter a name in the Search box.

      The system searches user names and group names.
   c. Select the user or group.
7. Select a role from the **Role** drop-down menu.
8. Decide whether to propagate the permissions by selecting the **Propagate to children** check box.

   If you assign a global permission and do not select **Propagate to children**, the users or groups associated with this permission do not have access to the objects in the hierarchy. They only have access to some global functionality such as creating roles.
9. Click **OK**.

Permissions on Tag Objects

In the vCenter Server object hierarchy, tag objects are not children of vCenter Server but are created at the vCenter Server top level. In environments with multiple vCenter Server instances, tag objects are shared across vCenter Server instances. Permissions for tag objects work differently than permissions for other objects in the vCenter Server object hierarchy.

Only Global Permissions or Permissions Assigned to the Tag Object Apply

If you grant permissions to a user on a vCenter Server inventory object, such as a virtual machine, that user can perform the tasks associated with the permission. However, the user cannot perform tag operations on the object.

For example, if you grant the **Assign vSphere Tag** privilege to user Dana on host TPA, that permission does not affect whether Dana can assign tags on host TPA. Dana must have the **Assign vSphere Tag** privilege at the top level, that is, a global permission, or must have the privilege for the tag object.

Table 2-1. How Global Permissions and Tag Object Permissions Affect What Users Can Do

Global Permission	Tag-Level Permission	vCenter Server Object-Level Permission	Effective Permission
No tagging privileges assigned.	Dana has **Assign or Unassign vSphere Tag** privileges for the tag.	Dana has **Delete vSphere Tag** privileges on ESXi host TPA.	Dana has **Assign or Unassign vSphere Tag** privileges for the tag.
Dana has **Assign or Unassign vSphere Tag** privileges.	No privileges assigned for the tag.	Dana has **Delete vSphere Tag** privileges on ESXi host TPA.	Dana has **Assign or Unassign vSphere Tag** global privileges. That includes privileges at the tag level.
No tagging privileges assigned.	No privileges assigned for the tag.	Dana has **Assign or Unassign vSphere Tag** privileges on ESXi host TPA.	Dana does not have tagging privileges on any object, including host TPA.

Global Permissions Complement Tag Object Permissions

Global permissions, that is, permissions that are assigned on the top-level object, complement permissions on tag objects when the permissions on the tag objects are more restrictive. The vCenter Server permissions do not affect the tag objects.

For example, assume that you assign the **Delete vSphere Tag** privilege to user Robin at the top level by using global permissions. For the tag Production, you do not assign the **Delete vSphere Tag** privilege to Robin. In that case, Robin has the privilege for the tag Production because Robin has the global permission, which propagates from the top level. You cannot restrict privileges unless you modify the global permission.

Table 2-2. Global Permissions Complement Tag-Level Permissions

Global Permission	Tag-Level Permission	Effective Permission
Robin has **Delete vSphere Tag** privileges	Robin does not have **Delete vSphere Tag** privileges for the tag.	Robin has **Delete vSphere Tag** privileges.
No tagging privileges assigned	Robin does not have **Delete vSphere Tag** privileges assigned for the tag.	Robin does not have **Delete vSphere Tag** privileges

Tag-Level Permissions Can Extend Global Permissions

You can use tag-level permissions to extend global permissions. That means users can have both a global permission and a tag-level permission on a tag.

Note This behavior is different from how vCenter Server privileges are inherited. In vCenter Server, permissions defined for a child object always override the permissions that are propagated from parent objects.

Table 2-3. Global Permissions Extend Tag-Level Permissions

Global Permission	Tag-Level Permission	Effective Permission
Lee has **Assign or Unassign vSphere Tag** privilege.	Lee has **Delete vSphere Tag** privilege.	Lee has the **Assign vSphere Tag** privilege and the **Delete vSphere Tag** privilege for the tag.
No tagging privileges assigned.	Lee has **Delete vSphere Tag** privilege assigned for the tag.	Lee has the **Delete vSphere Tag** privilege for the tag.

Using Roles to Assign Privileges

A role is a predefined set of privileges. Privileges define rights to perform actions and read properties. For example, the Virtual Machine Administrator role allows a user to read and change virtual machine attributes.

When you assign permissions, you pair a user or group with a role and associate that pairing with an inventory object. A single user or group can have different roles for different objects in the inventory.

For example, assume that you have two resource pools in your inventory, Pool A and Pool B. You can assign group Sales the Virtual Machine User role on Pool A, and the Read Only role on Pool B. With these assignments, the users in group Sales can turn on virtual machines in Pool A, but can only view virtual machines in Pool B.

vCenter Server provides system roles and sample roles by default.

System roles

System roles are permanent. You cannot edit the privileges associated with these roles.

Sample roles

VMware provides sample roles for certain frequently performed combination of tasks. You can clone, modify, or remove these roles.

Note To avoid losing the predefined settings in a sample role, clone the role first and make modifications to the clone. You cannot reset the sample to its default settings.

Users can schedule tasks only if they have a role that includes privileges to perform that task at the time the task is created.

Note Changes to roles and privileges take effect immediately, even if the users involved are logged in. The exception is searches, where changes take effect after the user has logged out and logged back in.

Custom Roles in vCenter Server and ESXi

You can create custom roles for vCenter Server and all objects that it manages, or for individual hosts.

vCenter Server Custom Roles (Recommended)

Create custom roles by using the role-editing facilities in the vSphere Client to create privilege sets that match your needs.

ESXi Custom Roles

You can create custom roles for individual hosts by using a CLI or the VMware Host Client. See the *vSphere Single Host Management - VMware Host Client* documentation. Custom host roles are not accessible from vCenter Server.

If you manage ESXi hosts through vCenter Server, do not maintain custom roles in both the host and vCenter Server. Define roles at the vCenter Server level.

When you manage a host using vCenter Server, the permissions associated with that host are created through vCenter Server and stored on vCenter Server. If you connect directly to a host, only the roles that are created directly on the host are available.

Note When you add a custom role and do not assign any privileges to it, the role is created as a Read Only role with three system-defined privileges: **System.Anonymous**, **System.View**, and **System.Read**. These privileges are not visible in the vSphere Client but are used to read certain properties of some managed objects. All the predefined roles in vCenter Server contain these three system-defined privileges. See the *vSphere Web Services API* documentation for more information.

Create a vCenter Server Custom Role

To suit the access control needs of your environment, you can create vCenter Server custom roles. You can create a role or clone an existing role.

You can create or edit a role on a vCenter Server system that is part of the same vCenter Single Sign-On domain as other vCenter Server systems. The VMware Directory Service (vmdir) propagates the role changes that you make to all other vCenter Server systems in the group. Assignments of roles to specific users and objects are not shared across vCenter Server systems.

Prerequisites

Verify that you are logged in as a user with Administrator privileges.

Procedure

1 Log in to the vCenter Server by using the vSphere Client.

2 Select **Administration** and click **Roles** in the **Access Control** area.

3 Create the role:

Option	Description
To create a role	Click **New**.
To create the role by cloning	Select a role, and click **Clone**.

See vCenter Server System Roles for more information.

4 Enter a name for the new role.

5 Select and deselect privileges for the role.

Scroll the privilege categories and select all privileges or a subset of privileges for that category. You can show all, selected, or unselected categories. You can also show all, selected, or unselected privileges.

See Chapter 16 Defined Privileges for more information.

Note When creating a cloned role, you cannot change privileges. To change privileges, select the cloned role and click **Edit**.

6 Click **Add**.

What to do next

You can now create permissions by selecting an object and assigning the role to a user or group for that object.

vCenter Server System Roles

A role is a predefined set of privileges. When you add permissions to an object, you pair a user or group with a role. vCenter Server includes some default system roles, which you cannot change.

vCenter Server provides a few default roles. You cannot change the privileges associated with the default roles. The default roles are organized as a hierarchy. Each role inherits the privileges of the previous role. For example, the Administrator role inherits the privileges of the Read Only role.

To view the privileges associated with a default role, navigate to the role in the vSphere Client (**Menu > Administration > Roles**) and click the **Privileges** tab.

To view all the vSphere privileges and descriptions, see Chapter 16 Defined Privileges.

The vCenter Server role hierarchy also includes several sample roles. You can clone a sample role to create a similar role.

If you create a role, it does not inherit privileges from any of the system roles.

Administrator Role

Users with the Administrator role for an object are allowed to view and perform all actions on the object. This role also includes all privileges of the Read Only role. If you have the Administrator role on an object, you can assign privileges to individual users and groups.

If you are acting in the Administrator role in vCenter Server, you can assign privileges to users and groups in the default vCenter Single Sign-On identity source. See the *vSphere Authentication* documentation for supported identity services.

By default, the administrator@vsphere.local user has the Administrator role on both vCenter Single Sign-On and vCenter Server after installation. That user can then associate other users with the Administrator role on vCenter Server.

Read Only Role

Users with the Read Only role for an object are allowed to view the state of the object and details about the object. For example, users with this role can view virtual machine, host, and resource pool attributes, but cannot view the remote console for a host. All actions through the menus and toolbars are disallowed.

No Access Role

Users with the No Access role for an object cannot view or change the object in any way. New users and groups are assigned this role by default. You can change the role on an object-by-object basis.

The administrator of the vCenter Single Sign-On domain, administrator@vsphere.local by default, the root user, and vpxuser are assigned the Administrator role by default. Other users are assigned the No Access role by default.

Best practice is to create a user at the root level and assign the Administrator role to that user. After creating a named user with Administrator privileges, you can remove the root user from any permissions or change its role to No Access.

Best Practices for Roles and Permissions

Follow best practices for roles and permissions to maximize the security and manageability of your vCenter Server environment.

Follow these best practices when configuring roles and permissions in your vCenter Server environment:

- Where possible, assign a role to a group rather than individual users.
- Grant permissions only on the objects where they are needed, and assign privileges only to users or groups that must have them. Use the minimum number of permissions to make it easier to understand and manage your permissions structure.

- If you assign a restrictive role to a group, check that the group does not contain the Administrator user or other users with administrative privileges. Otherwise, you might unintentionally restrict administrators' privileges in the parts of the inventory hierarchy where you have assigned that group the restrictive role.
- Use folders to group objects. For example, to grant modify permission on one set of hosts and view permission on another set of hosts, place each set of hosts in a folder.
- Use caution when adding a permission to the root vCenter Server objects. Users with privileges at the root level have access to global data on vCenter Server, such as roles, custom attributes, vCenter Server settings.
- Consider enabling propagation when you assign permissions to an object. Propagation ensures that new objects in the object hierarchy inherit permissions. For example, you can assign a permission to a virtual machine folder and enable propagation to ensure that the permission applies to all VMs in the folder.
- Use the No Access role to mask specific areas of the hierarchy. The No Access role restricts access for the users or groups with that role.
- Changes to licenses propagate to all linked vCenter Server systems in the same vCenter Single Sign-On domain.
- License propagation happens even if the user does not have privileges on all vCenter Server systems.

Required Privileges for Common Tasks

Many tasks require permissions on multiple objects in the inventory. If the user who attempts to perform the task only has privileges on one object, the task cannot complete successfully.

The following table lists common tasks that require more than one privilege. You can add permissions to inventory objects by pairing a user with one of the predefined roles or with multiple privileges. If you expect that you assign a set of privileges multiple times, create custom roles.

Refer to the *vSphere Web Services API Reference* documentation to learn how operations in the vSphere Client user interface map to API calls, and what privileges are required to perform operations. For example, the API documentation for the `AddHost_Task(addHost)` method specifies that the **Host.Inventory.AddHostToCluster** privilege is required to add a host to a cluster.

If the task that you want to perform is not in this table, the following rules explain where you must assign permissions to allow particular operations:

- Any operation that consumes storage space requires the **Datastore.Allocate Space** privilege on the target datastore, and the privilege to perform the operation itself. You must have these privileges, for example, when creating a virtual disk or taking a snapshot.

- Moving an object in the inventory hierarchy requires appropriate privileges on the object itself, the source parent object (such as a folder or cluster), and the destination parent object.
- Each host and cluster has its own implicit resource pool that contains all the resources of that host or cluster. Deploying a virtual machine directly to a host or cluster requires the **Resource.Assign Virtual Machine to Resource Pool** privilege.

Table 2-4. Required Privileges for Common Tasks

Task	Required Privileges	Applicable Role
Create a virtual machine	On the destination folder or data center: ■ **Virtual machine.Inventory.Create new** ■ **Virtual machine.Configuration.Add new disk** (if creating a new virtual disk) ■ **Virtual machine.Configuration.Add existing disk** (if using an existing virtual disk) ■ **Virtual machine.Configuration.Configure Raw device** (if using an RDM or SCSI pass-through device)	Administrator
	On the destination host, cluster, or resource pool: **Resource.Assign virtual machine to resource pool**	Resource pool administrator or Administrator
	On the destination datastore or the folder that contains the datastore: **Datastore.Allocate space**	Datastore Consumer or Administrator
	On the network that the virtual machine will be assigned to: **Network.Assign network**	Network Consumer or Administrator
Power on a virtual machine	On the data center in which the virtual machine is deployed: **Virtual machine.Interaction.Power On**	Virtual Machine Power User or Administrator
	On the virtual machine or folder of virtual machines: **Virtual machine.Interaction.Power On**	
Deploy a virtual machine from a template	On the destination folder or data center: ■ **Virtual machine.Inventory.Create from existing** ■ **Virtual machine.Configuration.Add new disk**	Administrator
	On a template or folder of templates: **Virtual machine.Provisioning.Deploy template**	Administrator
	On the destination host, cluster or resource pool: ■ **Resource.Assign virtual machine to resource pool** ■ **vApp.Import**	Administrator
	On the destination datastore or folder of datastores: **Datastore.Allocate space**	Datastore Consumer or Administrator
	On the network that the virtual machine will be assigned to: **Network.Assign network**	Network Consumer or Administrator

Table 2-4. Required Privileges for Common Tasks (continued)

Task	Required Privileges	Applicable Role
Take a virtual machine snapshot	On the virtual machine or a folder of virtual machines: **Virtual machine.Snapshot management.Create snapshot**	Virtual Machine Power User or Administrator
Move a virtual machine into a resource pool	On the virtual machine or folder of virtual machines: ■ **Resource.Assign virtual machine to resource pool** ■ **Virtual machine.Inventory.Move**	Administrator
	On the destination resource pool: **Resource.Assign virtual machine to resource pool**	Administrator
Install a guest operating system on a virtual machine	On the virtual machine or folder of virtual machines: ■ **Virtual machine.Interaction.Answer question** ■ **Virtual machine.Interaction.Console interaction** ■ **Virtual machine.Interaction.Device connection** ■ **Virtual machine.Interaction.Power Off** ■ **Virtual machine.Interaction.Power On** ■ **Virtual machine.Interaction.Reset** ■ **Virtual machine .Interaction.Configure CD media** (if installing from a CD) ■ **Virtual machine .Interaction.Configure floppy media** (if installing from a floppy disk) ■ **Virtual machine.Interaction.VMware Tools install**	Virtual Machine Power User or Administrator
	On a datastore that contains the installation media ISO image: **Datastore.Browse datastore** (if installing from an ISO image on a datastore) On the datastore to which you upload the installation media ISO image: ■ **Datastore.Browse datastore** ■ **Datastore.Low level file operations**	Virtual Machine Power User or Administrator
Migrate a virtual machine with vMotion	On the virtual machine or folder of virtual machines: ■ **Resource.Migrate powered on virtual machine** ■ **Resource.Assign Virtual Machine to Resource Pool** (if destination is a different resource pool from the source)	Resource Pool Administrator or Administrator
	On the destination host, cluster, or resource pool (if different from the source): **Resource.Assign virtual machine to resource pool**	Resource Pool Administrator or Administrator
Cold migrate (relocate) a virtual machine	On the virtual machine or folder of virtual machines: ■ **Resource.Migrate powered off virtual machine** ■ **Resource.Assign virtual machine to resource pool** (if destination is a different resource pool from the source)	Resource Pool Administrator or Administrator
	On the destination host, cluster, or resource pool (if different from the source): **Resource.Assign virtual machine to resource pool**	Resource Pool Administrator or Administrator

Table 2-4. Required Privileges for Common Tasks (continued)

Task	Required Privileges	Applicable Role
	On the destination datastore (if different from the source): **Datastore.Allocate space**	Datastore Consumer or Administrator
Migrate a virtual machine with Storage vMotion	On the virtual machine or folder of virtual machines: **Resource.Migrate powered on virtual machine**	Resource Pool Administrator or Administrator
	On the destination datastore: **Datastore.Allocate space**	Datastore Consumer or Administrator
Move a host into a cluster	On the host: **Host.Inventory.Add host to cluster**	Administrator
	On the destination cluster: ■ **Host.Inventory.Add host to cluster** ■ **Host.Inventory.Modify cluster**	Administrator
Add a single host to a data center by using the vSphere Client, or add a single host to a cluster by using PowerCLI or API (leveraging the addHost API)	On the host: **Host.Inventory.Add host to cluster**	Administrator
	On the cluster: ■ **Host.Inventory.Modify cluster** ■ **Host.Inventory.Add host to cluster**	Administrator
	On the data center: **Host.Inventory.Add standalone host**	Administrator
Add multiple hosts to a cluster	On the cluster: ■ **Host.Inventory.Modify cluster** ■ **Host.Inventory.Add host to cluster**	Administrator
	On the parent data center of the cluster (with propagate): ■ **Host.Inventory.Add standalone host** ■ **Host.Inventory.Move host** ■ **Host.Inventory.Modify cluster** ■ **Host.Configuration.Maintenance**	Administrator
Encrypt a virtual machine	Encryption tasks are possible only in environments that include vCenter Server. In addition, the ESXi host must have encryption mode enabled for most encryption tasks. The user who performs the task must have the appropriate privileges. A set of **Cryptographic Operations** privileges allows fine-grained control. See Prerequisites and Required Privileges for Encryption Tasks.	Administrator

Securing ESXi Hosts

3

The ESXi hypervisor architecture has many built-in security features such as CPU isolation, memory isolation, and device isolation. You can configure additional features such as lockdown mode, certificate replacement, and smart card authentication for enhanced security.

An ESXi host is also protected with a firewall. You can open ports for incoming and outgoing traffic as needed, but should restrict access to services and ports. Using the ESXi lockdown mode and limiting access to the ESXi Shell can further contribute to a more secure environment. ESXi hosts participate in the certificate infrastructure. Hosts are provisioned with certificates that are signed by the VMware Certificate Authority (VMCA) by default.

See the VMware white paper *Security of the VMware vSphere Hypervisor* for additional information on ESXi security.

Note ESXi is not built upon the Linux kernel or a commodity Linux distribution. It uses its own VMware specialized and proprietary kernel and software tools, delivered as a self-contained unit, and does not contain applications and components from Linux distributions.

Read the following topics next:

General ESXi Security Recommendations

To protect an ESXi host against an unauthorized intrusion and misuse, VMware imposes constraints on several parameters, settings, and activities. You can loosen the constraints to meet your configuration needs. If you do, make sure that you are working in a trusted environment and take other security measures.

Built-In Security Features

Risks to the hosts are mitigated as follows:

- ESXi Shell and SSH interfaces are disabled by default. Keep these interfaces disabled unless you are performing troubleshooting or support activities. For day-to-day activities, use the vSphere Client, where activity is subject to role-based access control and modern access control methods.
- Only a limited number of firewall ports are open by default. You can explicitly open additional firewall ports that are associated with specific services.
- ESXi runs only services that are essential to managing its functions. The distribution is limited to the features required to run ESXi.
- By default, all ports that are not required for management access to the host are closed. Open ports if you need additional services.
- By default, weak ciphers are disabled and communications from clients are secured by SSL. The exact algorithms used for securing the channel depend on the SSL handshake. Default certificates created on ESXi use PKCS#1 SHA-256 with RSA encryption as the signature algorithm.
- An internal web service is used by ESXi to support access by Web clients. The service has been modified to run only functions that a Web client requires for administration and monitoring. As a result, ESXi is not vulnerable to web service security issues reported in broader use.
- VMware monitors all security alerts that can affect ESXi security and issues a security patch if needed. You can subscribe to the VMware Security Advisories and Security Alerts mailing list to receive security alerts. See the webpage at http://lists.vmware.com/mailman/listinfo/security-announce.
- Insecure services such as FTP and Telnet are not installed, and the ports for these services are closed by default.
- To protect hosts from loading drivers and applications that are not cryptographically signed, use UEFI Secure boot. Enabling Secure Boot is done at the system BIOS. No additional configuration changes are required on the ESXi host, for example, to disk partitions. See UEFI Secure Boot for ESXi Hosts.
- If your ESXi host has a TPM 2.0 chip, enable and configure the chip in the system BIOS. Working together with Secure Boot, TPM 2.0 provides enhanced security and trust assurance rooted in hardware. See Securing ESXi Hosts with Trusted Platform Module.

Additional Security Measures

Consider the following recommendations when evaluating host security and administration.

Limit access

If you enable access to the Direct Console User Interface (DCUI), the ESXi Shell, or SSH, enforce strict access security policies.

The ESXi Shell has privileged access to certain parts of the host. Provide only trusted users with ESXi Shell login access.

Do not access managed hosts directly

Use the vSphere Client to administer ESXi hosts that are managed by a vCenter Server. Do not access managed hosts directly with the VMware Host Client, and do not change managed hosts from the DCUI.

If you manage hosts with a scripting interface or API, do not target the host directly. Instead, target the vCenter Server system that manages the host and specify the host name.

Use DCUI only for troubleshooting

Access the host from the DCUI or the ESXi Shell as the root user only for troubleshooting. To administer your ESXi hosts, use one of the GUI clients, or one of the VMware CLIs or APIs. See *ESXCLI Concepts and Examples* at https://code.vmware.com/. If you use the ESXi Shell or SSH, limit the accounts that have access and set timeouts.

Use only VMware sources to upgrade ESXi components

The host runs several third-party packages to support management interfaces or tasks that you must perform. VMware only supports upgrades to these packages that come from a VMware source. If you use a download or patch from another source, you might compromise management interface security or functions. Check third-party vendor sites and the VMware knowledge base for security alerts.

Note Follow the VMware security advisories at http://www.vmware.com/security/.

Advanced System Settings

Advanced system settings control aspects of ESXi behavior, such as logging, system resources, and security.

The following table presents some of the important ESXi advanced system settings for security. To view all the advanced system settings, consult either the vSphere Client (**Host > Configure > System > Advanced System Settings)** or the API for a given release.

Table 3-1. Partial List of Security Advanced System Settings

Advanced System Setting	Description	Default Value
Annotations.WelcomeMessage	Displays a welcome message in the Host Client prior to login, or in the DCUI on the default screen. In the DCUI, the welcome message replaces some text, such as the host IP address.	(Empty)
Config.Etc.issue	Displays a banner during an SSH login session. Use a trailing newline for best results.	(Empty)
Config.Etc.motd	Displays the message of the day upon SSH login.	(Empty)
Config.HostAgent.vmacore.soap.sessionTimeout	Sets the idle time in minutes before the system automatically logs out a VIM API. A value of 0 (zero) deactivates the idle time. This setting applies only to new sessions.	30 (minutes)
Mem.MemEagerZero	Activates zeroing the user world and the guest memory pages in the VMkernel operating systems (including the VMM process) after a virtual machine exit. The default value (0) uses lazy zeroing. A value of 1 uses eager zeroing.	0 (deactivated)
Security.AccountLockFailures	Sets the maximum number of failed login attempts before the system locks a user's account. For example, to lock the account on the fifth login failure, set this value to 4. A value of 0 (zero) deactivates account locking. For implementation reasons, some login mechanisms count unexpectedly: ■ VIM logins (including the VMware Host Client) and ESXCLI reflect the exact number of failed logins. ■ SSH connections count as a login attempt when displaying a password prompt, and undo that count on successful login. This behavior is normal for challenge and response communications. ■ CGI logins double-count login failures. Caution Due to this problem, a user can be locked out faster than the number of failed logins when using the CGI interface.	5

Table 3-1. Partial List of Security Advanced System Settings (continued)

Advanced System Setting	Description	Default Value
Security.AccountUnlockTime	Sets the number of seconds that a user is locked out. Any login attempt within the specified lock timeout restarts the lock timeout.	900 (15 minutes)
Security.PasswordHistory	Sets the number of passwords to remember for each user. This setting prevents duplicate or similar passwords.	0
Security.PasswordMaxDays	Sets the maximum number of days between password changes.	99999
Security.PasswordQualityControl	Changes the required length and the character class requirement or allow pass phrases in the `Pam_passwdqc` configuration. You can use special characters in passwords. You can have password lengths of at least 15 characters. The default setting requires three character classes and a minimum length of seven characters. If implementing the DoD Annex, you can combine the `similar=deny` option plus a minimum password length to enforce a requirement that passwords are sufficiently different. The password history setting is only enforced for passwords changed through the VIM `LocalAccountManager.changePassword` API. To change the password requires that the user have administrator permission. The PasswordQualityControl setting, with a PasswordMaxDays setting, satisfies the requirements of the DoD Annex: `min=disabled,disabled,disabled,disabled,15 similar=deny`	retry=3 min=disabled,disabled,disabled,7,7
UserVars.DcuiTimeOut	Sets the idle time in seconds before the system automatically logs out the DCUI. A value of 0 (zero) deactivates the timeout.	600 (10 minutes)

Table 3-1. Partial List of Security Advanced System Settings (continued)

Advanced System Setting	Description	Default Value
UserVars.ESXiShellInteractiveTimeOut	Sets the idle time in seconds before the system automatically logs out an interactive shell. This setting takes effect for new sessions only. A value of 0 (zero) deactivates the idle time. Applies to both the DCUI and the SSH shell.	0
UserVars.ESXiShellTimeOut	Sets the time in seconds a login shell waits for login. A value of 0 (zero) deactivates the timeout. Applies to both the DCUI and the SSH shell.	0
UserVars.HostClientSessionTimeout	Sets the idle time in seconds before the system automatically logs out the Host Client. A value of 0 (zero) deactivates the idle time.	900 (15 minutes)
UserVars.HostClientWelcomeMessage	Displays a welcome message in the Host Client upon login. The message is displayed following login as a “hint”.	(Empty)

Configure ESXi Hosts with Host Profiles

Host profiles allow you to set up standard configurations for your ESXi hosts and automate compliance to these configuration settings. Host profiles allow you to control many aspects of host configuration including memory, storage, networking, and so on.

You can configure host profiles for a reference host from the vSphere Client and apply the host profile to all hosts that share the characteristics of the reference host. You can also use host profiles to monitor hosts for host configuration changes. See *vSphere Host Profiles* documentation.

You can attach the host profile to a cluster to apply it to all hosts in the cluster.

Procedure

1. Set up the reference host to specification and create a host profile.
2. Attach the profile to a host or cluster.
3. Apply the host profile of the reference host to other hosts or clusters.

Use Scripts to Manage Host Configuration Settings

In environments with many hosts, managing hosts with scripts is faster and less error prone than managing the hosts from the vSphere Client.

vSphere includes several scripting languages for host management. See *ESXCLI Documentation* and *vSphere API/SDK Documentation* for reference information and programming tips, and VMware Communities for additional tips about scripted management. The vSphere Administrator documentation focuses on using the vSphere Client for management.

VMware PowerCLI

VMware PowerCLI is a Windows PowerShell interface to the vSphere API. VMware PowerCLI includes PowerShell cmdlets for administering vSphere components.

VMware PowerCLI includes hundreds of cmdlets, a set of sample scripts, and a function library for management and automation. See https://developer.vmware.com/powercli.

ESXCLI

ESXCLI includes a set of commands for managing ESXi hosts and virtual machines. See *ESXCLI Documentation*.

You can also use one of the scripting interfaces to the vSphere Automation SDK such as the vSphere Automation SDK for Python.

Procedure

1. Create a custom role that has limited privileges.

   For example, consider creating a role that has a set of privileges for managing hosts but no privileges for managing virtual machines, storage, or networking. If the script you want to use only extracts information, you can create a role with read-only privileges for the host.

2. From the vSphere Client, create a service account and assign it the custom role.

   You can create multiple custom roles with different levels of access if you want access to certain hosts to be fairly limited.

3 Write scripts to perform parameter checking or modification, and run them.

For example, you can check or set the shell interactive timeout of a host as follows:

Language	Commands
ESXCLI	`esxcli <conn_options> system settings advanced get / UserVars/ESXiShellTimeOut`
	`esxcli --formatter=csv --format-param=fields="Path,Int Value" system settings advanced list | grep /UserVars/ ESXiShellTimeOut`

PowerCLI	`#List UserVars.ESXiShellInteractiveTimeOut for each host Get-VMHost | Select Name, @{N="UserVars.ESXiShellInteractiveTimeOut";E={$_ | Get-AdvancedSetting -Name UserVars.ESXiShellInteractiveTimeOut | Select -ExpandProperty Value}}`
	
	`# Set UserVars.ESXiShellTimeOut to 900 on all hosts Get-VMHost | Foreach { Get-AdvancedSetting -Entity $_ -Name UserVars.ESXiShellInteractiveTimeOut | Set-AdvancedSetting -Value 900 }`

4 In large environments, create roles with different access privileges and group hosts into folders according to the tasks that you want to perform. You can then run scripts over different folders from different service accounts.

5 Verify that the changes happened after you run the command.

ESXi Passwords and Account Lockout

For ESXi hosts, you must use a password with predefined requirements. You can change the required length and character class requirement or allow pass phrases using the `Security.PasswordQualityControl` advanced option. You can also set the number of passwords to remember for each user using the `Security.PasswordHistory` advanced option.

Note The default requirements for ESXi passwords can change from one release to the next. You can check and change the default password restrictions by using the `Security.PasswordQualityControl` advanced option.

ESXi Passwords

ESXi enforces password requirements for access from the Direct Console User Interface, the ESXi Shell, SSH, or the VMware Host Client.

- By default, you must include a mix of at least three from the following four character classes: lowercase letters, uppercase letters, numbers, and special characters such as underscore or dash when you create a password.

- By default, password length is at least 7 characters and less than 40.
- Passwords must not contain a dictionary word or part of a dictionary word.

Note An uppercase character that begins a password does not count toward the number of character classes used. A number that ends a password does not count toward the number of character classes used. A dictionary word used inside a password reduces the overall password strength.

Example ESXi Passwords

The following password candidates illustrate potential passwords if the option is set as follows.

```
retry=3 min=disabled,disabled,disabled,7,7
```

With this setting, a user is prompted up to three times (retry=3) for a new password that is not sufficiently strong or if the password was not entered correctly twice. Passwords with one or two character classes and pass phrases are not allowed, because the first three items are disabled. Passwords from three- and four-character classes require seven characters. See the `pam_passwdqc` man page for details on other options, such as `max`, `passphrase`, and so on.

With these settings, the following passwords are allowed.

- xQaTEhb!: Contains eight characters from three character classes.
- xQaT3#A: Contains seven characters from four character classes.

The following password candidates do not meet requirements.

- Xqat3hi: Begins with an uppercase character, reducing the effective number of character classes to two. The minimum number of required character classes is three.
- xQaTEh2: Ends with a number, reducing the effective number of character classes to two. The minimum number of required character classes is three.

ESXi Pass Phrase

Instead of a password, you can also use a pass phrase. However, pass phrases are disabled by default. You can change the default setting and other settings by using the `Security.PasswordQualityControl` advanced option from the vSphere Client.

For example, you can change the option to the following.

```
retry=3 min=disabled,disabled,16,7,7
```

This example allows pass phrases of at least 16 characters and at least three words.

Changing Default Password Restrictions

You can change the default restriction on passwords or pass phrases by using the `Security.PasswordQualityControl` advanced option for your ESXi host. See *vCenter Server and Host Management* documentation for information on setting ESXi advanced options.

You can change the default, for example, to require a minimum of 15 characters and a minimum number of four words (`passphrase=4`), as follows:

```
retry=3 min=disabled,disabled,15,7,7 passphrase=4
```

See the man page for `pam_passwdqc` for details.

Note Not all possible combinations of password options have been tested. Perform testing after you change the default password settings.

This example sets the password complexity requirement to require eight characters from four character classes that enforce a significant password difference, a remembered history of five passwords, and a 90 day rotation policy:

```
min=disabled,disabled,disabled,disabled,8 similar=deny
```

Set the `Security.PasswordHistory` option to 5 and the `Security.PasswordMaxDays` option to 90.

ESXi Account Lockout Behavior

Account locking is supported for access through SSH and through the vSphere Web Services SDK. The Direct Console Interface (DCUI) and the ESXi Shell do not support account lockout. By default, a maximum of five failed attempts is allowed before the account is locked. The account is unlocked after 15 minutes by default.

Configuring Login Behavior

You can configure the login behavior for your ESXi host with the following advanced options:

- `Security.AccountLockFailures`. Maximum number of failed login attempts before a user's account is locked. Zero deactivates account locking.
- `Security.AccountUnlockTime`. Number of seconds that a user is locked out.
- `Security.PasswordHistory`. Number of passwords to remember for each user. Zero deactivates password history.

See the *vCenter Server and Host Management* documentation for information on setting ESXi advanced options.

Cryptographic Key Generation

ESXi generates several asymmetric keys for normal operation. The Transport Layer Security (TLS) key secures communication with the ESXi host using the TLS protocol. The SSH key secures communication with the ESXi host using the SSH protocol.

Transport Layer Security Key

The Transport Layer Security (TLS) key secures communication with the host using the TLS protocol. Upon first boot, the ESXi host generates the TLS key as a 2048-bit RSA key. Currently, ESXi does not implement automatic generation of ECDSA keys for TLS. The TLS private key is not intended to be serviced by the administrator.

The TLS key resides at the following non-persistent location:

```
/etc/vmware/ssl/rui.key
```

The TLS public key (including intermediate certificate authorities) resides at the following non-persistent location as an X.509 v3 certificate :

```
/etc/vmware/ssl/rui.crt
```

When you use vCenter Server with your ESXi hosts, vCenter Server generates a CSR automatically, signs it using the VMware Certificate Authority (VMCA), and generates the certificate. When you add an ESXi host to vCenter Server, vCenter Server installs that resulting certificate on the ESXi host.

The default TLS certificate is self-signed, with a subjectAltName field matching the host name at installation. You can install a different certificate, for example, to make use of a different subjectAltName or to include a particular Certificate Authority (CA) in the verification chain. See Replacing ESXi SSL Certificates and Keys.

You can also use the VMware Host Client to replace the certificate. See *vSphere Single Host Management - VMware Host Client*.

SSH Key

The SSH key secures communication with the ESXi host using the SSH protocol. Upon first boot, the system generates a nistp256 ECDSA key, and the SSH keys as 2048-bit RSA keys. The SSH server is deactivated by default. SSH access is intended primarily for troubleshooting purposes. The SSH keys are not intended to be serviced by the administrator. Logging in through SSH requires administrative privileges equivalent to full host control. To enable SSH access, see Enable Access to the ESXi Shell.

The SSH public keys reside at the following location:

```
/etc/ssh/ssh_host_rsa_key.pub
```

```
/etc/ssh/ssh_host_ecdsa_key.pub
```

The SSH private keys reside at the following location:

```
/etc/ssh/ssh_host_rsa_key
```

```
/etc/ssh/ssh_host_ecdsa_key
```

TLS Cryptographic Key Establishment

Configuration of TLS cryptographic key establishment is governed by choice of TLS cipher suites, which select one of the RSA-based key transports (as specified in NIST Special Publication 800-56B) or ECC-based key agreements using ephemeral Ecliptic Curve Diffie Hellman (ECDH) (as specified in NIST Special Publication 800-56A).

SSH Cryptographic Key Establishment

Configuration of SSH cryptographic key establishment is governed by the SSHD configuration. ESXi provides a default configuration that permits RSA-based key transport (as specified in NIST Special Publication 800-56B), ephemeral Diffie Hellman (DH) (as specified in NIST Special Publication 800-56A) key agreement, and ephemeral Ecliptic Curve Diffie Hellman (ECHD) (as specified in NIST Special Publication 800-56A). The SSHD configuration is not intended to be serviced by the administrator.

SSH Security

ESXi Shell and SSH interfaces are disabled by default. Keep these interfaces disabled unless you are performing troubleshooting or support activities. For day-to-day activities, use the vSphere Client, where activity is subject to role-based access control and modern access control methods.

The SSH configuration in ESXi uses the following settings:

Version 1 SSH protocol disabled

VMware does not support Version 1 SSH protocol and uses Version 2 protocol exclusively. Version 2 eliminates certain security problems present in Version 1 and provides you with a safe way to communicate with the management interface.

Improved cipher strength

SSH supports only 256-bit and 128-bit AES ciphers for your connections.

These settings are designed to provide solid protection for the data you transmit to the management interface through SSH. You cannot change these settings.

ESXi SSH Keys

SSH keys can restrict, control, and secure access to an ESXi host. An SSH key can allow a trusted user or script to log in to a host without entering a password.

You can copy the SSH key to the host by using the `vifs` command. You can also use HTTPS PUT to copy the SSK key to the host.

Instead of generating the keys externally and uploading them, you can create the keys on the ESXi host and download them. See the VMware knowledge base article at http://kb.vmware.com/kb/1002866.

Enabling SSH and adding SSH keys to the host has inherent risks. Weigh the potential risk of exposing a user name and password against the risk of intrusion by a user who has a trusted key.

Upload an SSH Key Using a vifs Command

If you decide that you want to use authorized keys to log in to a host with SSH, you can upload authorized keys with a `vifs` command.

Note Because authorized keys allow SSH access without requiring user authentication, consider carefully whether you want to use SSH keys in your environment.

Authorized keys allow you to authenticate remote access to a host. When users or scripts try to access a host with SSH, the key provides authentication without a password. With authorized keys, you can automate authentication, which is useful when you write scripts to perform routine tasks.

You can upload the following types of SSH keys to a host.

- Authorized keys file for the root user
- RSA key
- RSA public key

Starting with the vSphere 6.0 Update 2 release, DSS/DSA keys are no longer supported.

Important Do not modify the `/etc/ssh/sshd_config` file. If you do, you make a change that the host daemon (`hostd`) knows nothing about.

Procedure

- At the command line or an administration server, use the `vifs` command to upload the SSH key to an appropriate location on the ESXi host.

```
vifs --server hostname --username username --put filename /host/ssh_host_dsa_key_pub
```

Type of key	Location
Authorized key files for the root user	`/host/ssh_root_authorized_keys` You must have full administrator privileges to upload this file.
RSA keys	`/host/ssh_host_rsa_key`
RSA public keys	`/host/ssh_host_rsa_key_pub`

Upload an SSH Key Using HTTPS PUT

You can use authorized keys to log in to a host with SSH. You can upload authorized keys with HTTPS PUT.

Authorized keys allow you to authenticate remote access to a host. When users or scripts try to access a host with SSH, the key provides authentication without a password. With authorized keys you can automate authentication, which is useful when you write scripts to perform routine tasks.

You can upload the following types of SSH keys to a host using HTTPS PUT:

- Authorized keys file for root user
- DSA key
- DSA public key
- RSA key
- RSA public key

Important Do not modify the `/etc/ssh/sshd_config` file.

Procedure

1. In your upload application, open the key file.
2. Publish the file to the following locations.

Type of key	Location
Authorized key files for the root user	`https://hostname_or_IP_address/host/ssh_root_authorized_keys` You must have full administrator privileges on the host to upload this file.
DSA keys	`https://hostname_or_IP_address/host/ssh_host_dsa_key`
DSA public keys	`https://hostname_or_IP_address/host/ssh_host_dsa_key_pub`
RSA keys	`https://hostname_or_IP_address/host/ssh_host_rsa_key`
RSA public keys	`https://hostname_or_IP_address/host/ssh_host_rsa_key_pub`

PCI and PCIe Devices and ESXi

Using the VMware DirectPath I/O feature to pass through a PCI or PCIe device to a virtual machine results in a potential security vulnerability. The vulnerability can be triggered when buggy or malicious code, such as a device driver, is running in privileged mode in the guest OS. Industry-standard hardware and firmware do not currently have sufficient error containment support to protect ESXi hosts from the vulnerability.

Use PCI or PCIe passthrough to a virtual machine only if a trusted entity owns and administers the virtual machine. You must be sure that this entity does not to attempt to crash or exploit the host from the virtual machine.

Your host might be compromised in one of the following ways.

- The guest OS might generate an unrecoverable PCI or PCIe error. Such an error does not corrupt data, but can crash the ESXi host. Such errors might occur because of bugs or incompatibilities in the hardware devices that are being passed through. Other reasons for errors include problems with drivers in the guest OS.
- The guest OS might generate a Direct Memory Access (DMA) operation that causes an IOMMU page fault on the ESXi host. This operation might be the result of a DMA operation

that targets an address outside the virtual machine memory. On some machines, host firmware configures IOMMU faults to report a fatal error through a non-maskable interrupt (NMI). This fatal error causes the ESXi host to crash. This problem might occur because of problems with the drivers in the guest OS.

- If the operating system on the ESXi host is not using interrupt remapping, the guest OS might inject a spurious interrupt into the ESXi host on any vector. ESXi currently uses interrupt remapping on Intel platforms where it is available. Interrupt mapping is part of the Intel VT-d feature set. ESXi does not use interrupt mapping on AMD platforms. A false interrupt can result in a crash of the ESXi host. Other ways to exploit these false interrupts might exist in theory.

Disable the Managed Object Browser

The managed object browser (MOB) provides a way to explore the VMkernel object model. However, attackers can use this interface to perform malicious configuration changes or actions because it is possible to change the host configuration by using the MOB. Use the MOB only for debugging, and ensure that it is disabled in production systems.

The MOB is disabled by default. However, for certain tasks, for example when extracting the old certificate from a system, you have to use the MOB. You can enable and disable the MOB as follows.

Procedure

1. Browse to the host in the vSphere Client inventory.
2. Click **Configure**.
3. Under System, click **Advanced System Settings**.
4. Check the value of **Config.HostAgent.plugins.solo.enableMob**, and click **Edit** to change it as appropriate.

   Do not use `vim-cmd` from the ESXi Shell.

ESXi Networking Security Recommendations

Isolation of network traffic is essential to a secure ESXi environment. Different networks require a different access and level of isolation.

Your ESXi host uses several networks. Use appropriate security measures for each network, and isolate traffic for specific applications and functions. For example, ensure that VMware vSphere® vMotion® traffic does not travel over networks where virtual machines are located. Isolation prevents snooping. Having separate networks is also recommended for performance reasons.

- vSphere infrastructure networks are used for features such as vSphere vMotion, VMware vSphere Fault Tolerance, VMware vSAN, and storage. Isolate these networks for their specific functions. It is often not necessary to route these networks outside a single physical server rack.

- A management network isolates client traffic, command-line interface (CLI) or API traffic, and third-party software traffic from other traffic. In general, the management network is accessible only by system, network, and security administrators. To secure access to the management network, use a bastion host or a virtual private network (VPN). Strictly control access within this network.
- Virtual machine traffic can flow over one or many networks. You can enhance the isolation of virtual machines by using virtual firewall solutions that set firewall rules at the virtual network controller. These settings travel with a virtual machine as it migrates from host to host within your vSphere environment.

Modifying ESXi Web Proxy Settings

When you modify Web proxy settings, you have several encryption and user security guidelines to consider.

Note Restart the host process after making any changes to host directories or authentication mechanisms.

- Do not set up certificates that use a password or pass phrases. ESXi does not support Web proxies that use passwords or pass phrases, also known as encrypted keys. If you set up a Web proxy that requires a password or pass phrase, ESXi processes cannot start correctly.
- To support encryption for user names, passwords, and packets, SSL is enabled by default for vSphere Web Services SDK connections. If you want to configure these connections so that they do not encrypt transmissions, disable SSL for your vSphere Web Services SDK connection by switching the connection from HTTPS to HTTP.

  Consider disabling SSL only if you created a fully trusted environment for these clients, where firewalls are in place and transmissions to and from the host are fully isolated. Disabling SSL can improve performance, because you avoid the overhead required to perform encryption.
- To protect against misuse of ESXi services, most internal ESXi services are accessible only through port 443, the port used for HTTPS transmission. Port 443 acts as a reverse proxy for ESXi. You can see a list of services on ESXi through an HTTP welcome page, but you cannot directly access the Storage Adapters services without proper authorization.

  You can change this configuration so that individual services are directly accessible through HTTP connections. Do not make this change unless you are using ESXi in a fully trusted environment.
- When you upgrade your environment, the certificate remains in place.

vSphere Auto Deploy Security Considerations

When you use vSphere Auto Deploy, pay careful attention to networking security, boot image security, and potential password exposure through host profiles to protect your environment.

Networking Security

Secure your network just as you secure the network for any other PXE-based deployment method. vSphere Auto Deploy transfers data over SSL to prevent casual interference and snooping. However, the authenticity of the client or of the Auto Deploy server is not checked during a PXE boot.

You can greatly reduce the security risk of Auto Deploy by completely isolating the network where Auto Deploy is used.

Boot Image and Host Profile Security

The boot image that the vSphere Auto Deploy server downloads to a machine can have the following components.

- The VIB packages that the image profile consists of are always included in the boot image.
- The host profile and host customization are included in the boot image if Auto Deploy rules are set up to provision the host with a host profile or host customization.
  - The administrator (root) password and user passwords that are included with host profile and host customization are hashed with SHA-512.
  - Any other passwords associated with profiles are in the clear. If you set up Active Directory by using host profiles, the passwords are not protected.

    Use the vSphere Authentication Proxy to avoid exposing the Active Directory passwords. If you set up Active Directory using host profiles, the passwords are not protected.
- The host's public and private SSL key and certificate are included in the boot image.

Control Access for CIM-Based Hardware Monitoring Tools

The Common Information Model (CIM) system provides an interface that enables hardware-level management from remote applications using a set of standard APIs. To ensure that the CIM interface is secure, provide only the minimum access necessary to these remote applications. If you provision a remote application with a root or Administrator account, and if the application is compromised, the virtual environment can be compromised.

CIM is an open standard that defines a framework for agent-less, standards-based monitoring of hardware resources for ESXi hosts. This framework consists of a CIM object manager, often called a CIM broker, and a set of CIM providers.

CIM providers support management access to device drivers and underlying hardware. Hardware vendors, including server manufacturers and hardware device vendors, can write providers that monitor and manage their devices. VMware writes providers that monitor server hardware, ESXi storage infrastructure, and virtualization-specific resources. These providers run inside the ESXi host and are lightweight and focused on specific management tasks. The CIM broker takes information from all CIM providers and presents it to the outside world using standard APIs. The most common API is WS-MAN.

Do not provide root credentials to remote applications that access the CIM interface. Instead, create a less-privileged vSphere user account for these applications and use the VIM API ticket function to issue a sessionId (called a "ticket") to this less-privileged user account to authenticate to CIM. If the account has been granted permission to obtain CIM tickets, the VIM API can then supply the ticket to CIM. These tickets are then supplied as both the user ID and password to any CIM-XML API call. See the `AcquireCimServicesTicket()` method for more information.

The CIM service starts when you install a third-party CIM VIB, for example, when you run the `esxcli software vib install -n` *`VIBname`* command.

If you must enable the CIM service manually, run the following command:

```
esxcli system wbem set -e true
```

If necessary, you can disable wsman (WSManagement Service) so that only the CIM service is running:

```
esxcli system wbem set -W false
```

To confirm that wsman is disabled, run the following command:

```
esxcli system wbem get
…
WSManagement PID: 0
WSManagement Service: false
```

For more information about ESXCLI commands, see *ESXCLI Documentation*. For more information about enabling the CIM service, see the VMware knowledge base article at https://kb.vmware.com/kb/1025757.

Procedure

1. Create a non-root vSphere user account for CIM applications.

   See the topic on adding vCenter Single Sign-On users in *vSphere Authentication*. The required vSphere privilege for the user account is **Host.CIM.Interaction**.

2. Use the vSphere API SDK of your choice to authenticate the user account to vCenter Server. Then call `AcquireCimServicesTicket()` to return a ticket to authenticate with ESXi as an administrator-level account using CIM-XML port 5989 or WS-Man port 433 APIs.

   See *vSphere Web Services API Reference* for more information.

3. Renew the ticket every two minutes as needed.

Certificate Management for ESXi Hosts

The VMware Certificate Authority (VMCA) provisions each new ESXi host with a signed certificate that has VMCA as the root certificate authority by default. Provisioning happens when the host is added to vCenter Server explicitly or as part of installation or upgrade to ESXi 6.0 or later.

You can view and manage ESXi certificates from the vSphere Client and by using the `vim.CertificateManager` API in the vSphere Web Services SDK. You cannot view or manage ESXi certificates by using certificate management CLIs that are available for managing vCenter Server certificates.

Certificates in vSphere 6.0 and Later

When ESXi and vCenter Server communicate, they use TLS for almost all management traffic.

In vSphere 6.0 and later, vCenter Server supports the following certificate modes for ESXi hosts.

Table 3-2. Certificate Modes for ESXi Hosts

Certificate Mode	Description
VMware Certificate Authority (default)	Use this mode if VMCA provisions all ESXi hosts, either as the top-level CA or as an intermediate CA. By default, VMCA provisions ESXi hosts with certificates. In this mode, you can refresh and renew certificates from the vSphere Client.
Custom Certificate Authority	Use this mode if you want to use only custom certificates that are signed by a third-party or enterprise CA. In this mode, you are responsible for managing the certificates. You cannot refresh and renew certificates from the vSphere Client. **Note** Unless you change the certificate mode to Custom Certificate Authority, VMCA might replace custom certificates, for example, when you select **Renew** in the vSphere Client.
Thumbprint Mode	vSphere 5.5 used thumbprint mode, and this mode is still available as a fallback option for vSphere 6.x. In this mode, vCenter Server checks that the certificate is formatted correctly, but does not check the validity of the certificate. Even expired certificates are accepted. Do not use this mode unless you encounter problems that you cannot resolve with one of the other two modes. Some vCenter 6.x and later services might not work correctly in thumbprint mode.

Certificate Expiration

You can view information about certificate expiration for certificates that are signed by VMCA or a third-party CA in the vSphere Client. You can view the information for all hosts that are managed by a vCenter Server or for individual hosts. A yellow alarm is raised if the certificate is in the **Expiring Shortly** state (less than eight months). A red alarm is raised if the certificate is in the **Expiration Imminent** state (less than two months).

ESXi Provisioning and VMCA

When you boot an ESXi host from installation media, the host initially has an autogenerated certificate. When the host is added to the vCenter Server system, it is provisioned with a certificate that is signed by VMCA as the root CA.

The process is similar for hosts that are provisioned with Auto Deploy. However, because those hosts do not store any state, the signed certificate is stored by the Auto Deploy server in its local certificate store. The certificate is reused during subsequent boots of the ESXi hosts. An Auto Deploy server is part of any embedded deployment or vCenter Server system.

If VMCA is not available when an Auto Deploy host boots the first time, the host first attempts to connect. If the host cannot connect, it cycles through shutdown and reboot until VMCA becomes available and the host can be provisioned with a signed certificate.

Required Privileges for ESXi Certificate Management

For certificate management for ESXi hosts, you must have the **Certificates.Manage Certificates** privilege. You can set that privilege from the vSphere Client.

Host Name and IP Address Changes

A host name or IP address change might affect whether vCenter Server considers a host certificate valid. How you added the host to vCenter Server affects whether manual intervention is necessary. Manual intervention means that you either reconnect the host, or you remove the host from vCenter Server and add it back.

Table 3-3. When Host Name or IP Address Changes Require Manual Intervention

Host added to vCenter Server using...	Host name changes	IP address changes
Host name	vCenter Server connectivity problem. Manual intervention required.	No intervention required.
IP address	No intervention required.	vCenter Server connectivity problem. Manual intervention required.

Host Upgrades and Certificates

If you upgrade an ESXi host to ESXi 6.5 or later, the upgrade process replaces the self-signed (thumbprint) certificates with VMCA-signed certificates. If the ESXi host uses custom certificates, the upgrade process retains those certificates even if those certificates are expired or invalid.

The recommended upgrade workflow depends on the current certificates.

Host Provisioned with Thumbprint Certificates

If your host is currently using thumbprint certificates, it is automatically assigned VMCA certificates as part of the upgrade process.

Note You cannot provision legacy hosts with VMCA certificates. You must upgrade those hosts to ESXi 6.5 or later.

Host Provisioned with Custom Certificates

If your host is provisioned with custom certificates, usually third-party CA-signed certificates, those certificates remain in place during upgrade. Change the certificate mode to **Custom** to ensure that the certificates are not replaced accidentally during a certificate refresh later.

Note If your environment is in VMCA mode, and you refresh the certificates from the vSphere Client, any existing certificates are replaced with certificates that are signed by VMCA.

Going forward, vCenter Server monitors the certificates and displays information, for example, about certificate expiration, in the vSphere Client.

Hosts Provisioned with Auto Deploy

Hosts that are being provisioned by Auto Deploy are always assigned new certificates when they are first booted with ESXi 6.5 or later software. When you upgrade a host that is provisioned by Auto Deploy, the Auto Deploy server generates a certificate signing request (CSR) for the host and submits it to VMCA. VMCA stores the signed certificate for the host. When the Auto Deploy server provisions the host, it retrieves the certificate from VMCA and includes it as part of the provisioning process.

You can use Auto Deploy with custom certificates.

See Use Custom Certificates with Auto Deploy.

Certificate Mode Switch Workflows

Starting with vSphere 6.0, ESXi hosts are provisioned with certificates by VMCA by default. You can instead use custom certificate mode or, for debugging purposes, the legacy thumbprint mode. In most cases, mode switches are disruptive and not necessary. If you do require a mode switch, review the potential impact before you start.

In vSphere 6.0 and later, vCenter Server supports the following certificate modes for ESXi hosts.

Certificate Mode	Description
VMware Certificate Authority (default)	By default, the VMware Certificate Authority is used as the CA for ESXi host certificates. VMCA is the root CA by default, but it can be set up as the intermediary CA to another CA. In this mode, users can manage certificates from the vSphere Client. Also used if VMCA is a subordinate certificate.
Custom Certificate Authority	Some customers might prefer to manage their own external certificate authority. In this mode, customers are responsible for managing the certificates and cannot manage them from the vSphere Client.
Thumbprint Mode	vSphere 5.5 used thumbprint mode, and this mode is still available as a fallback option for vSphere 6.0. Do not use this mode unless you encounter problems with one of the other two modes that you cannot resolve. Some vCenter 6.0 and later services might not work correctly in thumbprint mode.

Using Custom ESXi Certificates

If your company policy requires that you use a different root CA than VMCA, you can switch the certificate mode in your environment after careful planning. The workflow is as follows.

1 Obtain the certificates that you want to use.

2 Place the host or hosts into maintenance mode and disconnect them from vCenter Server.

3 Add the custom CA's root certificate to VECS.

4 Deploy the custom CA certificates to each host and restart services on that host.

5 Switch to Custom CA mode. See Change the Certificate Mode.

6 Connect the host or hosts to the vCenter Server system.

Switching from Custom CA Mode to VMCA Mode

If you are using custom CA mode and decide that using VMCA works better in your environment, you can perform the mode switch after careful planning. The workflow is as follows.

1 Remove all hosts from the vCenter Server system.

2 On the vCenter Server system, remove the third-party CA's root certificate from VECS.

3 Switch to VMCA mode. See Change the Certificate Mode.

4 Add the hosts to the vCenter Server system.

Note Any other workflow for this mode switch might result in unpredictable behavior.

Retaining Thumbprint Mode Certificates During Upgrade

The switch from VMCA mode to thumbprint mode might be necessary if you encounter problems with the VMCA certificates. In thumbprint mode, the vCenter Server system checks only whether a certificate exists and is formatted correctly, and does not check whether the certificate is valid. See Change the Certificate Mode for instructions.

Switching from Thumbprint Mode to VMCA Mode

If you use thumbprint mode and you want to start using VMCA-signed certificates, the switch requires some planning. The workflow is as follows.

1 Remove all hosts from the vCenter Server system.

2 Switch to VMCA certificate mode. See Change the Certificate Mode.

3 Add the hosts to the vCenter Server system.

Note Any other workflow for this mode switch might result in unpredictable behavior.

Switching from Custom CA Mode to Thumbprint Mode

If you are encountering problems with your custom CA, consider switching to thumbprint mode temporarily. The switch works seamlessly if you follow the instructions in Change the Certificate Mode. After the mode switch, the vCenter Server system checks only the format of the certificate and no longer checks the validity of the certificate itself.

Switching from Thumbprint Mode to Custom CA Mode

If you set your environment to thumbprint mode during troubleshooting, and you want to start using custom CA mode, you must first generate the required certificates. The workflow is as follows.

1 Remove all hosts from the vCenter Server system.

2 Add the custom CA root certificate to TRUSTED_ROOTS store on VECS on the vCenter Server system. See Update the vCenter Server TRUSTED_ROOTS Store (Custom Certificates).

3 For each ESXi host:

   a Deploy the custom CA certificate and key.

   b Restart services on the host.

4 Switch to custom mode. See Change the Certificate Mode.

5 Add the hosts to the vCenter Server system.

ESXi Certificate Default Settings

When a host is added to a vCenter Server system, vCenter Server sends a Certificate Signing Request (CSR) for the host to VMCA. Most of the default values are well suited for many situations, but company-specific information can be changed.

You can change many of the default settings using the vSphere Client. Consider changing the organization, and location information. See Change Certificate Default Settings.

Table 3-4. ESXi CSR Settings

Parameter	Default Value	Advanced Option
Key Size	2048	N.A.
Key Algorithm	RSA	N.A.
Certificate Signature Algorithm	sha256WithRSAEncryption	N.A.
Common Name	Name of the host if the host was added to vCenter Server by host name. IP address of the host if the host was added to vCenter Server by IP address.	N.A.
Country	US	vpxd.certmgmt.certs.cn.country
Email address	vmca@vmware.com	vpxd.certmgmt.certs.cn.email
Locality (City)	Palo Alto	vpxd.certmgmt.certs.cn.localityName
Organization Unit Name	VMware Engineering	vpxd.certmgmt.certs.cn.organizationalUnitName
Organization Name	VMware	vpxd.certmgmt.certs.cn.organizationName
State or province	California	vpxd.certmgmt.certs.cn.state
Number of days the certificate is valid.	1825	vpxd.certmgmt.certs.daysValid
Hard threshold for the certificate expiration. vCenter Server raises a red alarm when this threshold is reached.	30 days	vpxd.certmgmt.certs.hardThreshold
Poll interval for vCenter Server certificate validity checks.	5 days	vpxd.certmgmt.certs.pollIntervalDays
Soft threshold for the certificate expiration. vCenter Server raises an event when this threshold is reached.	240 days	vpxd.certmgmt.certs.softThreshold
Mode that vCenter Server users to determine whether existing certificates are replaced. Change this mode to retain custom certificates during upgrade. See Host Upgrades and Certificates.	vmca You can also specify thumbprint or custom. See Change the Certificate Mode.	vpxd.certmgmt.mode

Change Certificate Default Settings

When a host is added to a vCenter Server system, vCenter Server sends a Certificate Signing Request (CSR) for the host to VMCA. You can change some of the default settings in the CSR using the vCenter Server Advanced Settings in the vSphere Client.

See ESXi Certificate Default Settings for a list of default settings. Some of the defaults cannot be changed.

Procedure

1 In the vSphere Client, select the vCenter Server system that manages the hosts.

2 Click **Configure**, and click **Advanced Settings**.

3 Click **Edit Settings**.

4 Click the **Filter** icon in the Name column, and in the Filter box, enter **`vpxd.certmgmt`** to display only certificate management parameters.

5 Change the value of the existing parameters to follow your company policy and click **Save**.

   The next time you add a host to vCenter Server, the new settings are used in the CSR that vCenter Server sends to VMCA and in the certificate that is assigned to the host.

What to do next

Changes to certificate metadata only affect new certificates. If you want to change the certificates of hosts that are already managed by the vCenter Server system, you can disconnect and reconnect the hosts or renew the certificates.

View Certificate Expiration Information for Multiple ESXi Hosts

If you are using ESXi 6.0 and later, you can view the certificate status of all hosts that are managed by your vCenter Server system. The display allows you to determine whether any of the certificates expire soon.

You can view certificate status information for hosts that are using VMCA mode and for hosts that are using custom mode in the vSphere Client. You cannot view certificate status information for hosts in thumbprint mode.

Procedure

1 Log in to the vCenter Server by using the vSphere Client.

2 Browse the inventory list and select the vCenter Server instance.

3 Select **Hosts & Clusters > Hosts**.

   By default, the Hosts display does not include the certificate status.

4 To show or hide columns, click the three-bar **Column Selector** in the lower left corner.

5 Select the **Certificate Valid To** check box, and scroll to the right if necessary to view the added column.

   The certificate information displays when the certificate expires.

If a host is added to vCenter Server or reconnected after a disconnect, vCenter Server renews the certificate if the status is Expired, Expiring, Expiring shortly, or Expiration imminent. The status is Expiring if the certificate is valid for less than eight months, Expiring shortly if the certificate is valid for less than two months, and Expiration imminent if the certificate is valid for less than one month.

6 (Optional) Deselect other columns to make it easier to see what you are interested in.

What to do next

Renew the certificates that are about to expire. See Renew or Refresh ESXi Certificates.

View Certificate Details for a Single ESXi Host

For ESXi 6.0 and later hosts that are in VMCA mode or custom mode, you can view certificate details from the vSphere Client. The information about the certificate can be helpful for debugging.

Procedure

1 Browse to the host in the vSphere Client inventory.

2 Click **Configure**.

3 Under **System**, click **Certificate**.

You can examine the following information. This information is available only in the single-host view.

Field	Description
Subject	The subject used during certificate generation.
Issuer	The issuer of the certificate.
Valid From	Date on which the certificate was generated.

Field	Description
Valid To	Date on which the certificate expires.
Status	Status of the certificate, one of the following. **Good** Normal operation. **Expiring** Certificate expires soon. **Expiring shortly** Certificate is eight months or less away from expiration (Default). **Expiration imminent** Certificate is two months or less away from expiration (Default). **Expired** Certificate is not valid because it expired.

Renew or Refresh ESXi Certificates

If VMCA assigns certificates to your ESXi hosts (6.0 and later), you can renew those certificates from the vSphere Client. You can also refresh all certificates from the TRUSTED_ROOTS store associated with vCenter Server.

You can renew your certificates when they are about to expire, or if you want to provision the host with a new certificate for other reasons. If you do not renew the certificate before it expires, disconnecting the host and reconnecting it causes vCenter Server to renew the certificate. The act of re-adding the host to vCenter Server reestablishes trust, and enables vCenter Server to unconditionally issue the renewed certificate.

By default, vCenter Server renews the certificates of a host with status Expired, Expiration imminent, or Expiring shortly, each time the host is added to the inventory, or reconnected.

Prerequisites

Verify the following:

- The ESXi hosts are connected to the vCenter Server system.
- There is proper time synchronization between the vCenter Server system and the ESXi hosts.
- DNS resolution works between the vCenter Server system and the ESXi hosts.
- The vCenter Server system's MACHINE_SSL_CERT and Trusted_Root certificates are valid and have not expired. See the VMware knowledge base article at https://kb.vmware.com/s/article/2111411.
- The ESXi hosts are not in maintenance mode.

Procedure

1 Browse to the host in the vSphere Client inventory.

2 Click **Configure**.

3 Under **System**, click **Certificate**.

You can view detailed information about the selected host's certificate.

4 Click **Renew** or **Refresh CA Certificates**.

Option	Description
Renew	Retrieves a fresh signed certificate for the host from VMCA.
Refresh CA Certificates	Pushes all certificates in the TRUSTED_ROOTS store in the vCenter Server VECS store to the host.

5 Click **Yes** to confirm.

Change the Certificate Mode

Use VMCA to provision the ESXi hosts in your environment unless corporate policy requires that you use custom certificates. To use custom certificates with a different root CA, you can edit the vCenter Server `vpxd.certmgmt.mode` advanced option. After the change, the hosts are no longer automatically provisioned with VMCA certificates when you refresh the certificates. You are responsible for the certificate management in your environment.

You can use the vCenter Server advanced settings to change to thumbprint mode or to custom CA mode. Use thumbprint mode only as a fallback option.

Procedure

1 In the vSphere Client, select the vCenter Server system that manages the hosts.

2 Click **Configure**, and under Settings, click **Advanced Settings**.

3 Click **Edit Settings**.

4 Click the **Filter** icon in the Name column, and in the Filter box, enter **`vpxd.certmgmt`** to display only certificate management parameters.

5 Change the value of `vpxd.certmgmt.mode` to **custom** if you intend to manage your own certificates, and to **thumbprint** if you temporarily want to use thumbprint mode, and click **Save**.

6 Restart the vCenter Server service.

See the *vCenter Server Configuration* documentation for information about restarting services.

Replacing ESXi SSL Certificates and Keys

Your company's security policy might require that you replace the default ESXi SSL certificate with a third-party CA-signed certificate on each host.

By default, vSphere components use the VMCA-signed certificate and key that are created during installation. If you accidentally delete the VMCA-signed certificate, remove the host from its vCenter Server system, and add it back. When you add the host, vCenter Server requests a new certificate from VMCA and provisions the host with it.

Replace VMCA-signed certificates with certificates from a trusted CA, either a commercial CA or an organizational CA, if your company policy requires it.

The default certificates are in the same location as the vSphere 5.5 certificates. You can replace the default certificates with trusted certificates in various ways.

Note You can also use the `vim.CertificateManager` and `vim.host.CertificateManager` managed objects in the vSphere Web Services SDK. See the vSphere Web Services SDK documentation.

After you replace the certificate, you have to update the TRUSTED_ROOTS store in VECS on the vCenter Server system that manages the host to ensure that the vCenter Server and the ESXi host have a trust relationship.

For detailed instructions about using CA-signed certificates for ESXi hosts, see Certificate Mode Switch Workflows.

Note If you are replacing SSL certificates on an ESXi host that is part of a vSAN cluster, follow the steps that are in the VMware knowledge base article at https://kb.vmware.com/s/article/56441.

What to read next

- Requirements for ESXi Certificate Signing Requests

  If you want to use an enterprise or third-party CA-signed certificate, or a subordinate CA-signed certificate, you have to send a Certificate Signing Request (CSR) to the CA.
- Replace the Default Certificate and Key from the ESXi Shell

  You can replace the default VMCA-signed ESXi certificates from the ESXi Shell.
- Replace a Default Certificate and Key with the vifs Command

  You can replace the default VMCA-signed ESXi certificates by using the `vifs` command.
- Replace a Default Certificate Using HTTPS PUT

  You can use third-party applications to upload certificates and key. Applications that support HTTPS PUT operations work with the HTTPS interface that is included with ESXi.

- Update the vCenter Server TRUSTED_ROOTS Store (Custom Certificates)

  If you set up your ESXi hosts to use custom certificates, you must update the `TRUSTED_ROOTS` store on the vCenter Server system that manages the hosts.

Requirements for ESXi Certificate Signing Requests

If you want to use an enterprise or third-party CA-signed certificate, or a subordinate CA-signed certificate, you have to send a Certificate Signing Request (CSR) to the CA.

Use a CSR with these characteristics:

- Key size: 2048 bits (minimum) to 16384 bits (maximum) (PEM encoded)
- PEM format. VMware supports PKCS8 and PKCS1 (RSA keys). When keys are added to VECS, they are converted to PKCS8.
- x509 version 3
- For root certificates, the CA extension must be set to true, and the cert sign must be in the list of requirements.
- SubjectAltName must contain DNS Name=<machine_FQDN>.
- CRT format
- Contains the following Key Usages: Digital Signature, Non Repudiation, Key Encipherment
- Start time of one day before the current time.
- CN (and SubjectAltName) set to the host name (or IP address) that the ESXi host has in the vCenter Server inventory.

vSphere does not support the following certificates.

- Certificates with wildcards.
- The algorithms md2WithRSAEncryption, md5WithRSAEncryption, RSASSA-PSS, dsaWithSHA1, ecdsa_with_SHA1, and sha1WithRSAEncryption are not supported.

For information about generating the CSR, see the VMware knowledge base article at https://kb.vmware.com/s/article/2113926.

Replace the Default Certificate and Key from the ESXi Shell

You can replace the default VMCA-signed ESXi certificates from the ESXi Shell.

Prerequisites

- If you want to use third-party CA-signed certificates, generate the certificate request, send it to the certificate authority, and store the certificates on each ESXi host.
- If necessary, enable the ESXi Shell or enable SSH traffic from the vSphere Client.
- All file transfers and other communications occur over a secure HTTPS session. The user who is used to authenticate the session must have the privilege **Host.Config.AdvancedConfig** on the host.

Procedure

1. Log in to the ESXi Shell, either directly from the DCUI or from an SSH client, as a user with administrator privileges.
2. In the directory `/etc/vmware/ssl`, rename the existing certificates using the following commands.

   ```
   mv rui.crt orig.rui.crt
   mv rui.key orig.rui.key
   ```

3. Copy the certificates that you want to use to `/etc/vmware/ssl`.
4. Rename the new certificate and key to `rui.crt` and `rui.key`.
5. Restart the host after you install the new certificate.

   Alternatively, you can put the host into maintenance mode, install the new certificate, use the Direct Console User Interface (DCUI) to restart the management agents, and set the host to exit maintenance mode.

What to do next

Update the vCenter Server TRUSTED_ROOTS store. See Update the vCenter Server TRUSTED_ROOTS Store (Custom Certificates).

Replace a Default Certificate and Key with the vifs Command

You can replace the default VMCA-signed ESXi certificates by using the `vifs` command.

Prerequisites

- If you want to use third-party CA-signed certificates, generate the certificate request, send it to the certificate authority, and store the certificates on each ESXi host.
- If necessary, enable the ESXi Shell or enable SSH traffic from the vSphere Client.
- All file transfers and other communications occur over a secure HTTPS session. The user who is used to authenticate the session must have the privilege **Host.Config.AdvancedConfig** on the host.

Procedure

1. Back up the existing certificates.
2. Generate a certificate request following the instructions from the certificate authority.

   See Requirements for ESXi Certificate Signing Requests.

3. When you have the certificate, use the `vifs` command to upload the certificate to the appropriate location on the host from an SSH connection to the host.

   ```
   vifs --server hostname --username username --put rui.crt /host/ssl_cert
   vifs --server hostname --username username --put rui.key /host/ssl_key
   ```

4 Restart the host.

Alternatively, you can put the host into maintenance mode, install the new certificate, use the Direct Console User Interface (DCUI) to restart the management agents, and set the host to exit maintenance mode.

What to do next

Update the vCenter Server TRUSTED_ROOTS store. See Update the vCenter Server TRUSTED_ROOTS Store (Custom Certificates).

Replace a Default Certificate Using HTTPS PUT

You can use third-party applications to upload certificates and key. Applications that support HTTPS PUT operations work with the HTTPS interface that is included with ESXi.

Prerequisites

- If you want to use third-party CA-signed certificates, generate the certificate request, send it to the certificate authority, and store the certificates on each ESXi host.
- If necessary, enable the ESXi Shell or enable SSH traffic from the vSphere Client.
- All file transfers and other communications occur over a secure HTTPS session. The user who is used to authenticate the session must have the privilege **Host.Config.AdvancedConfig** on the host.

Procedure

1 Back up the existing certificates.

2 Set up basic access authentication, in which you supply a Base64 encoded username and password, separated with a single colon (:). For more information, see https://en.wikipedia.org/wiki/Basic_access_authentication.

3 In your upload application, process each file as follows:

a Open the file.

b Publish the file to one of these locations.

Option	Description
Certificates	`https://hostname/host/ssl_cert`
Keys	`https://hostname/host/ssl_key`

The `/host/ssl_cert` and the `host/ssl_key` locations link to the certificate files in `/etc/vmware/ssl`.

4 Restart the host.

Alternatively, you can put the host into maintenance mode, install the new certificate, use the Direct Console User Interface (DCUI) to restart the management agents, and set the host to exit maintenance mode.

What to do next

Update the vCenter Server TRUSTED_ROOTS store. See Update the vCenter Server TRUSTED_ROOTS Store (Custom Certificates).

Update the vCenter Server TRUSTED_ROOTS Store (Custom Certificates)

If you set up your ESXi hosts to use custom certificates, you must update the `TRUSTED_ROOTS` store on the vCenter Server system that manages the hosts.

Prerequisites

Replace the certificates on each host with custom certificates.

Note This step is not required if the vCenter Server system is also running with custom certificates issued by the same CA as those installed on the ESXi hosts.

Procedure

1. Log in to the vCenter Server shell of the vCenter Server system that manages the ESXi hosts.
2. To add the new certificates to the `TRUSTED_ROOTS` store, run `dir-cli`, for example:

   ```
   /usr/lib/vmware-vmafd/bin/dir-cli trustedcert publish --cert path_to_RootCA
   ```

3. When prompted, provide the Single Sign-On Administrator credentials.
4. If your custom certificates are issued by an intermediate CA, you must also add the intermediate CA to the `TRUSTED_ROOTS` store on the vCenter Server, for example:

   ```
   /usr/lib/vmware-vmafd/bin/dir-cli trustedcert publish --cert path_to_intermediateCA
   ```

What to do next

Set certificate mode to Custom. If the certificate mode is VMCA, the default, and you perform a certificate refresh, your custom certificates are replaced with VMCA-signed certificates. See Change the Certificate Mode.

Use Custom Certificates with Auto Deploy

By default, the Auto Deploy server provisions each host with certificates that are signed by VMCA. You can set up the Auto Deploy server to provision all hosts with custom certificates that are not signed by VMCA. In that scenario, the Auto Deploy server becomes a subordinate certificate authority of your third-party CA.

Prerequisites

- Request a certificate from your CA. The certificate must meet these requirements.
  - Key size: 2048 bits (minimum) to 16384 bits (maximum) (PEM encoded)
  - PEM format. VMware supports PKCS8 and PKCS1 (RSA keys). When keys are added to VECS, they are converted to PKCS8.

- x509 version 3
- For root certificates, the CA extension must be set to true, and the cert sign must be in the list of requirements.
- SubjectAltName must contain DNS Name=<machine_FQDN>.
- CRT format
- Contains the following Key Usages: Digital Signature, Non Repudiation, Key Encipherment
- Start time of one day before the current time.
- CN (and SubjectAltName) set to the host name (or IP address) that the ESXi host has in the vCenter Server inventory.

- Name the certificate and the key files `rbd-ca.crt` and `rbd-ca.key`.

Procedure

1 Back up the default ESXi certificates.

The certificates are in the `/etc/vmware-rbd/ssl/` directory.

2 Stop the vSphere Authentication Proxy service.

Tool	Steps
vCenter Server Management Interface	a In a Web browser, go to the vCenter Server Management Interface, https://*vcenter-IP-address-or-FQDN*:5480. b Log in as root. The default root password is the password that you set while deploying the vCenter Server. c Click **Services**, and click the **VMware vSphere Authentication Proxy service**. d Click **Stop**.
CLI	`service-control --stop vmcam`

3 On the system where the Auto Deploy service runs, replace `rbd-ca.crt` and `rbd-ca.key` in `/etc/vmware-rbd/ssl/` with your custom certificate and key files.

4 On the system where the Auto Deploy service runs, run the following command to update the TRUSTED_ROOTS store inside the VECS to use your new certificates.

```
/usr/lib/vmware-vmafd/bin/dir-cli trustedcert publish --cert /etc/vmware-rbd/ssl/rbd-ca.crt
/usr/lib/vmware-vmafd/bin/vecs-cli force-refresh
```

5 Create a `castore.pem` file that contains what is in the TRUSTED_ROOTS store and place the file in the `/etc/vmware-rbd/ssl/` directory.

In custom mode, you are responsible for maintaining this file.

6 Change the ESXi certificate mode for the vCenter Server system to **custom**.

See Change the Certificate Mode.

7 Restart the vCenter Server service and start the Auto Deploy service.

Results

The next time you provision a host that is set up to use Auto Deploy, the Auto Deploy server generates a certificate. The Auto Deploy server uses the root certificate that you added to the TRUSTED_ROOTS store.

Note If you encounter problems with Auto Deploy after certificate replacement, see the VMware knowledge base article at http://kb.vmware.com/kb/2000988.

Restore ESXi Certificate and Key Files

When you replace a certificate on an ESXi host by using the vSphere Web Services SDK, the previous certificate and key are appended to a `.bak` file. You can restore previous certificates by moving the information in the `.bak` file to the current certificate and key files.

The host certificate and key are located in `/etc/vmware/ssl/rui.crt` and `/etc/vmware/ssl/rui.key`. When you replace a host certificate and key by using the vSphere Web Services SDK `vim.CertificateManager` managed object, the previous key and certificate are appended to the file `/etc/vmware/ssl/rui.bak`.

Note If you replace the certificate by using HTTP PUT, `vifs`, or from the ESXi Shell, the existing certificates are not appended to the `.bak` file.

Procedure

1 On the ESXi host, locate the file `/etc/vmware/ssl/rui.bak`.

The file has the following format.

```
#
# Host private key and certificate backup from 2014-06-20 08:02:49.961
#

-----BEGIN PRIVATE KEY-----
previous key
-----END PRIVATE KEY-----

-----BEGIN CERTIFICATE-----
previous cert
-----END CERTIFICATE-----
```

2 Copy the text starting with `-----BEGIN PRIVATE KEY-----` and ending with `-----END PRIVATE KEY-----` into the `/etc/vmware/ssl/rui.key` file.

Include `-----BEGIN PRIVATE KEY-----` and `-----END PRIVATE KEY-----`.

3 Copy the text between `-----BEGIN CERTIFICATE-----` and `-----END CERTIFICATE-----` into the `/etc/vmware/ssl/rui.crt` file.

   Include `-----BEGIN CERTIFICATE-----` and `-----END CERTIFICATE-----`.

4 Restart the ESXi host.

   Alternatively, you can put the host into maintenance mode and use the Direct Console User Interface (DCUI) to restart the management agents, and set the host to exit maintenance mode.

Customizing Hosts with the Security Profile

You can customize many of the essential security settings for your host through the Security Profile, Services, and Firewall panels available in the vSphere Client. The Security Profile is especially useful for single host management. If you are managing multiple hosts, consider using one of the CLIs or SDKs and automating the customization.

ESXi Firewall Configuration

ESXi includes a firewall that is enabled by default.

At installation time, the ESXi firewall is configured to block incoming and outgoing traffic, except traffic for services that are enabled in the security profile of the host.

As you open ports on the firewall, consider that unrestricted access to services running on an ESXi host can expose a host to outside attacks and unauthorized access. Reduce the risk by configuring the ESXi firewall to enable access only from authorized networks.

Note The firewall also allows Internet Control Message Protocol (ICMP) pings and communication with DHCP and DNS (UDP only) clients.

You can manage ESXi firewall ports as follows:

- Use **Configure > Firewall** for each host in the vSphere Client. See Manage ESXi Firewall Settings.
- Use ESXCLI commands from the command line or in scripts. See ESXi ESXCLI Firewall Commands.
- Use a custom VIB if the port you want to open is not included in the security profile.

  To install the custom VIB, you have to change the acceptance level of the ESXi host to CommunitySupported.

  Note If you engage VMware Technical Support to investigate a problem on an ESXi host with a CommunitySupported VIB installed, VMware Support can request you to uninstall this VIB. Such a request is a troubleshooting step to determine if that VIB is related to the problem being investigated.

The behavior of the NFS Client rule set (`nfsClient`) is different from other rule sets. When the NFS Client rule set is enabled, all outbound TCP ports are open for the destination hosts in the list of allowed IP addresses. See NFS Client Firewall Behavior for more information.

Manage ESXi Firewall Settings

You can configure incoming and outgoing firewall connections for a service or a management agent from the vSphere Client or at the command line.

This task describes how to use the vSphere Client to configure ESXi firewall settings. You can use the ESXi Shell or ESXCLI commands to configure ESXi at the command line to automate the firewall configuration. See *Getting Started with ESXCLI* for an introduction, and *ESXCLI Concepts and Examples* for examples of using ESXCLI to manipulate firewalls and firewall rules.

Note If different services have overlapping port rules, enabling one service might implicitly enable other services. You can specify which IP addresses are allowed to access each service on the host to avoid this problem.

Procedure

1. Log in to the vCenter Server by using the vSphere Client.
2. Browse to the host in the inventory.
3. Click **Configure**, then click **Firewall** under **System**.

   You can toggle between incoming and outgoing connections by clicking **Incoming** and **Outgoing**.
4. In the Firewall section, click **Edit**.
5. Select from one of the three service groups, **Ungrouped**, **Secure Shell**, and **Simple Network Management Protocol**.
6. Select the rule sets to enable, or deselect the rule sets to disable.
7. For some services, you can also manage service details by navigating to **Configure > Services** under System.

   For more information about starting, stopping, and restarting services, see Enable or Disable a Service.
8. For some services, you can explicitly specify IP addresses from which connections are allowed.

   See Add Allowed IP Addresses for an ESXi Host.
9. Click **OK**.

Add Allowed IP Addresses for an ESXi Host

By default, the firewall for each service allows access to all IP addresses. To restrict traffic, change each service to allow traffic only from your management subnet. You can also deselect some services if your environment does not use them.

To update the Allowed IP list for a service you can use the vSphere Client, ESXCLI, or PowerCLI. By default, all IP addresses are allowed for a service. This task describes how to use the vSphere Client. See the topic on managing the firewall in *ESXCLI Concepts and Examples* at https://code.vmware.com/ for instructions on using ESXCLI.

Procedure

1 Log in to the vCenter Server by using the vSphere Client.

2 Browse to the ESXi host.

3 Click **Configure**, then click **Firewall** under **System**.

You can toggle between incoming and outgoing connections by clicking **Incoming** and **Outgoing**.

4 In the Firewall section, click **Edit**.

5 Select from one of the three service groups, **Ungrouped**, **Secure Shell**, and **Simple Network Management Protocol**.

6 To display the Allowed IP Addresses section, expand a service.

7 In the Allowed IP Addresses section, deselect **Allow connections from any IP address** and enter the IP addresses of networks that are allowed to connect to the host.

Separate IP addresses with commas. You can use the following address formats:

- 192.168.0.0/24
- 192.168.1.2, 2001::1/64
- fd3e:29a6:0a81:e478::/64

8 Ensure that the service itself is selected.

9 Click **OK**.

10 Verify your change in the **Allowed IP addresses** column for the service.

Incoming and Outgoing Firewall Ports for ESXi Hosts

The vSphere Client and the VMware Host Client allow you to open and close firewall ports for each service or to allow traffic from selected IP addresses.

ESXi includes a firewall that is enabled by default. At installation time, the ESXi firewall is configured to block incoming and outgoing traffic, except traffic for services that are enabled in the host's security profile. For the list of supported ports and protocols in the ESXi firewall, see the VMware Ports and Protocols Tool™ at https://ports.vmware.com/.

The VMware Ports and Protocols Tool lists port information for services that are installed by default. If you install other VIBs on your host, additional services and firewall ports might become available. The information is primarily for services that are visible in the vSphere Client but the VMware Ports and Protocols Tool includes some other ports as well.

NFS Client Firewall Behavior

The NFS Client firewall rule set behaves differently than other ESXi firewall rule sets. ESXi configures NFS Client settings when you mount or unmount an NFS datastore. The behavior differs for different versions of NFS.

When you add, mount, or unmount an NFS datastore, the resulting behavior depends on the version of NFS.

NFS v3 Firewall Behavior

When you add or mount an NFS v3 datastore, ESXi checks the state of the NFS Client (`nfsClient`) firewall rule set.

- If the `nfsClient` rule set is disabled, ESXi enables the rule set and disables the Allow All IP Addresses policy by setting the `allowedAll` flag to `FALSE`. The IP address of the NFS server is added to the allowed list of outgoing IP addresses.
- If the `nfsClient` rule set is enabled, the state of the rule set and the allowed IP address policy are not changed. The IP address of the NFS server is added to the allowed list of outgoing IP addresses.

Note If you manually enable the `nfsClient` rule set or manually set the Allow All IP Addresses policy, either before or after you add an NFS v3 datastore to the system, your settings are overridden when the last NFS v3 datastore is unmounted. The `nfsClient` rule set is disabled when all NFS v3 datastores are unmounted.

When you remove or unmount an NFS v3 datastore, ESXi performs one of the following actions.

- If none of the remaining NFS v3 datastores are mounted from the server of the datastore being unmounted, ESXi removes the server's IP address from the list of outgoing IP addresses.
- If no mounted NFS v3 datastores remain after the unmount operation, ESXi disables the `nfsClient` firewall rule set.

NFS v4.1 Firewall Behavior

When you mount the first NFS v4.1 datastore, ESXi enables the `nfs41client` rule set and sets its `allowedAll` flag to TRUE. This action opens port 2049 for all IP addresses. Unmounting an NFS v4.1 datastore does not affect the firewall state. That is, the first NFS v4.1 mount opens port 2049 and that port remains enabled unless you close it explicitly.

ESXi ESXCLI Firewall Commands

If your environment includes multiple ESXi hosts, automate firewall configuration by using ESXCLI commands or the vSphere Web Services SDK.

Firewall Command Reference

You can use the ESXi Shell or ESXCLI commands to configure ESXi at the command line to automate a firewall configuration. To manipulate firewalls and firewall rules, see *Getting Started with ESXCLI* for an introduction, and *ESXCLI Concepts and Examples* for examples of using ESXCLI.

In ESXi 7.0 and later, access to the `service.xml` file, used to create custom firewall rules, is restricted. See VMware Knowledge Base article 2008226 for information about creating custom firewall rules using the `/etc/rc.local.d/local.sh` file.

Table 3-5. Firewall Commands

Command	Description
`esxcli network firewall get`	Return the enabled or disabled status of the firewall and lists default actions.
`esxcli network firewall set --default-action`	Set to true to set the default action to pass. Set to false to set the default action to drop.
`esxcli network firewall set --enabled`	Enable or disable the ESXi firewall.
`esxcli network firewall load`	Load the firewall module and the rule set configuration files.
`esxcli network firewall refresh`	Refresh the firewall configuration by reading the rule set files if the firewall module is loaded.
`esxcli network firewall unload`	Destroy filters and unload the firewall module.
`esxcli network firewall ruleset list`	List rule sets information.
`esxcli network firewall ruleset set --allowed-all`	Set to true to allow all access to all IPs. Set to false to use a list of allowed IP addresses.
`esxcli network firewall ruleset set --enabled --ruleset-id=<string>`	Set enabled to true to enable the specified ruleset. Set enabled to false to disable the specified ruleset.
`esxcli network firewall ruleset allowedip list`	List the allowed IP addresses of the specified rule set.
`esxcli network firewall ruleset allowedip add`	Allow access to the rule set from the specified IP address or range of IP addresses.
`esxcli network firewall ruleset allowedip remove`	Remove access to the rule set from the specified IP address or range of IP addresses.
`esxcli network firewall ruleset rule list`	List the rules of each ruleset in the firewall.

Customizing ESXi Services from the Security Profile

An ESXi host includes several services that are running by default. If your company policy allows it, you can disable services from the security profile, or enable services.

Enable or Disable a Service is an example of how to enable a service.

Note Enabling services affects the security of your host. Do not enable a service unless strictly necessary.

Available services depend on the VIBs that are installed on the ESXi host. You cannot add services without installing a VIB. Some VMware products, for example, vSphere HA, install VIBs on hosts and make services and the corresponding firewall ports available.

In a default installation, you can modify the status of the following services from the vSphere Client.

Table 3-6. ESXi Services in the Security Profile

Service	Default	Description
Direct Console UI	Running	The Direct Console User Interface (DCUI) service allows you to interact with an ESXi host from the local console host using text-based menus.
ESXi Shell	Stopped	The ESXi Shell is available from the Direct Console User Interface and includes a set of fully supported commands and a set of commands for troubleshooting and remediation. You must enable access to the ESXi Shell from the direct console of each system. You can enable access to the local ESXi Shell or access to the ESXi Shell with SSH.
SSH	Stopped	The host's SSH client service that allows remote connections through Secure Shell.
Load-Based Teaming Daemon	Running	Load-Based Teaming.
attestd	Stopped	vSphere Trust Authority Attestation Service.
kmxd	Stopped	vSphere Trust Authority Key Provider Service.
Active Directory Service	Stopped	When you configure ESXi for Active Directory, this service is started.
NTP Daemon	Stopped	Network Time Protocol daemon.
PC/SC Smart Card Daemon	Stopped	When you enable the host for smart card authentication, this service starts. See Configuring Smart Card Authentication for ESXi.
CIM Server	Running	Service that can be used by Common Information Model (CIM) applications.
SNMP Server	Stopped	SNMP daemon. See *vSphere Monitoring and Performance* for information on configuring SNMP v1, v2, and v3.
Syslog Server	Stopped	Syslog daemon. You can enable syslog from the Advanced System Settings in the vSphere Client. See *vCenter Server Installation and Setup*.
VMware vCenter Agent	Running	vCenter Server agent. Allows a vCenter Server to connect to an ESXi host. Specifically, vpxa is the communication conduit to the host daemon, which in turn communicates with the ESXi kernel.
X.Org Server	Stopped	X.Org Server. This optional feature is used internally for 3D graphics for virtual machines.

Enable or Disable a Service

You can enable or disable services from the vSphere Client.

After installation, certain services are running by default, while others are stopped. Sometimes, additional setup is necessary before a service becomes available in the UI. For example, the NTP service is a way of getting accurate time information, but this service only works when required ports are opened in the firewall.

Prerequisites

Connect to vCenter Server with the vSphere Client.

Procedure

1. Browse to a host in the inventory.
2. Click **Configure**, then click **Services** under System.
3. Select the service you want to change.
   a. Select **Restart**, **Start**, or **Stop** for a one-time change to the host's status.
   b. To change the status of the host across reboots, click **Edit Startup Policy** and select a policy.
   - **Start and stop with host**: The service starts shortly after the host starts, and closes shortly before the host shuts down. Much like **Start and stop with port usage**, this option means that the service regularly attempts to complete its tasks, such as contacting the specified NTP server. If the port was closed but is later opened, the client begins completing its tasks shortly thereafter.
   - **Start and stop manually**: The host preserves the user-determined service settings, regardless of whether ports are open or not. When a user starts the NTP service, that service is kept running if the host is powered on. If the service is started and the host is powered off, the service is stopped as part of the shutdown process. When the host is powered on, the service is started again, preserving the user-determined state.
   - **Start and stop with port usage**: The default setting for these services. If any port is open, the client attempts to contact the network resources for the service. If some ports are open, but the port for a particular service is closed, the attempt fails. If and when the applicable outgoing port is opened, the service begins completing its startup.

   Note These settings apply only to service settings that are configured through the UI or to applications that are created with the vSphere Web Services SDK. Configurations made through other means, such as from the ESXi Shell or with configuration files, are not affected by these settings.
4. Click **OK**.

Lockdown Mode

To increase the security of your ESXi hosts, you can put them in lockdown mode. In lockdown mode, operations must be performed through vCenter Server by default.

You can select normal lockdown mode or strict lockdown mode, which offer different degrees of lockdown. You can also use the Exception User list. Exception users do not lose their privileges when the host enters lockdown mode. Use the Exception User list to add the accounts of third-party solutions and external applications that need to access the host directly when the host is in lockdown mode. See Specify Lockdown Mode Exception Users.

Lockdown Mode Behavior

In lockdown mode, some services are disabled, and some services are accessible only to certain users.

Lockdown Mode Services for Different Users

When the host is running, available services depend on whether lockdown mode is enabled, and on the type of lockdown mode.

- In strict and normal lockdown mode, privileged users can access the host through vCenter Server, from the vSphere Client, or by using the vSphere Web Services SDK.
- Direct Console Interface behavior differs for strict lockdown mode and normal lockdown mode.
  - In strict lockdown mode, the Direct Console User Interface (DCUI) service is disabled.
  - In normal lockdown mode, accounts on the Exception User list can access the DCUI if they have administrator privileges. In addition, all users who are specified in the `DCUI.Access` advanced system setting can access the DCUI.
- If the ESXi Shell or SSH is enabled and the host is placed in lockdown mode, accounts on the Exception Users list who have administrator privileges can use these services. For all other users, ESXi Shell or SSH access is disabled. ESXi or SSH sessions for users who do not have administrator privileges are closed.

All access is logged for both strict and normal lockdown mode.

Table 3-7. Lockdown Mode Behavior

Service	Normal Mode	Normal Lockdown Mode	Strict Lockdown Mode
vSphere Web Services API	All users, based on permissions	vCenter (vpxuser) Exception users, based on permissions vCloud Director (vslauser, if available)	vCenter (vpxuser) Exception users, based on permissions vCloud Director (vslauser, if available)
CIM Providers	Users with administrator privileges on the host	vCenter (vpxuser) Exception users, based on permissions vCloud Director (vslauser, if available)	vCenter (vpxuser) Exception users, based on permissions vCloud Director (vslauser, if available)
Direct Console UI (DCUI)	Users with administrator privileges on the host, and users in the `DCUI.Access` advanced option	Users defined in the `DCUI.Access` advanced option Exception users with administrator privileges on the host	DCUI service is stopped.
ESXi Shell (if enabled) and SSH (if enabled)	Users with administrator privileges on the host	Users defined in the `DCUI.Access` advanced option Exception users with administrator privileges on the host	Users defined in the `DCUI.Access` advanced option Exception users with administrator privileges on the host

Users Logged In to the ESXi Shell When Lockdown Mode Is Enabled

Users might log in to the ESXi Shell or access the host through SSH before lockdown mode is enabled. In that case, users who are on the list of Exception Users and who have administrator privileges on the host remain logged in. The session is closed for all other users. Termination applies to both normal and strict lockdown mode.

Enable Lockdown Mode

Enable lockdown mode to require that all configuration changes go through vCenter Server. vSphere 6.0 and later supports normal lockdown mode and strict lockdown mode.

If you want to disallow all direct access to a host completely, you can select strict lockdown mode. Strict lockdown mode makes it impossible to access a host if the vCenter Server is unavailable and SSH and the ESXi Shell are disabled. See Lockdown Mode Behavior.

Procedure

1. Browse to the host in the vSphere Client inventory.
2. Click **Configure**.
3. Under System, select **Security Profile**.
4. In the Lockdown Mode panel, click **Edit**.

5 Click **Lockdown Mode** and select one of the lockdown mode options.

Option	Description
Normal	The host can be accessed through vCenter Server. Only users who are on the Exception Users list and have administrator privileges can log in to the Direct Console User Interface. If SSH or the ESXi Shell is enabled, access might be possible.
Strict	The host can only be accessed through vCenter Server. If SSH or the ESXi Shell is enabled, running sessions for accounts in the `DCUI.Access` advanced option and for Exception User accounts that have administrator privileges remain enabled. All other sessions are closed.

6 Click **OK**.

Disable Lockdown Mode

Disable lockdown mode to allow configuration changes from direct connections to the ESXi host. Leaving lockdown mode enabled results in a more secure environment.

You can disable lockdown mode as follows:

From the Graphical User Interface

Users can disable both normal lockdown mode and strict lockdown mode from the vSphere Client.

From the Direct Console User Interface

Users who can access the Direct Console User Interface on the ESXi host can disable normal lockdown mode. In strict lockdown mode, the Direct Console Interface service is stopped.

Procedure

1 Browse to a host in the vSphere Client inventory.

2 Click **Configure**.

3 Under System, select **Security Profile**.

4 In the Lockdown Mode panel, click **Edit**.

5 Click **Lockdown Mode** and select **Disabled** to disable lockdown mode.

6 Click **OK**.

Results

The system exits lockdown mode, vCenter Server displays an alarm, and an entry is added to the audit log.

Enable or Disable Normal Lockdown Mode from the Direct Console User Interface

You can enable and disable normal lockdown mode from the Direct Console User Interface (DCUI). You can enable and disable strict lockdown mode only from the vSphere Client.

When the host is in normal lockdown mode, the following accounts can access the Direct Console User Interface:

- Accounts in the Exception Users list who have administrator privileges on the host. The Exception Users list is meant for service accounts such as a backup agent.
- Users defined in the `DCUI.Access` advanced option for the host. This option can be used to enable access in a catastrophic failure.

User permissions are preserved when you enable lockdown mode. User permissions are restored when you disable lockdown mode from the Direct Console Interface.

Note If you upgrade a host that is in lockdown mode to ESXi version 6.0 without exiting lockdown mode, and if you exit lockdown mode after the upgrade, all permissions defined before the host entered lockdown mode are lost. The system assigns the administrator role to all users who are found in the `DCUI.Access` advanced option to guarantee that the host remains accessible.

To retain permissions, disable lockdown mode for the host from the vSphere Client before the upgrade.

Procedure

1. At the Direct Console User Interface of the host, press F2 and log in.
2. Scroll to the **Configure Lockdown Mode** setting and press Enter to toggle the current setting.
3. Press Esc until you return to the main menu of the Direct Console User Interface.

Specifying Accounts with Access Privileges in Lockdown Mode

You can specify service accounts that can access the ESXi host directly by adding them to the Exception Users list. You can specify a single user who can access the ESXi host in a catastrophic vCenter Server failure.

The vSphere version determines what different accounts can do by default when lockdown mode is enabled, and how you can change the default behavior.

- In vSphere 5.0 and earlier, only the root user can log in to the Direct Console User Interface on an ESXi host that is in lockdown mode.
- In vSphere 5.1 and later, you can add a user to the `DCUI.Access` advanced system setting for each host. The option is meant for a catastrophic failure of vCenter Server. Companies usually lock the password for the user with this access into a safe. A user in the `DCUI.Access` list does not need to have full administrative privileges on the host.

- In vSphere 6.0 and later, the `DCUI.Access` advanced system setting is still supported. In addition, vSphere 6.0 and later supports an Exception User list, which is for service accounts that have to log in to the host directly. Accounts with administrator privileges that are on the Exception Users list can log in to the ESXi Shell. In addition, those users can log in to a host's DCUI in normal lockdown mode and can exit lockdown mode.

  You specify Exception Users from the vSphere Client.

  Note Exception users are host local users or Active Directory users with privileges defined locally for the ESXi host. Users that are members of an Active Directory group lose their permissions when the host is in lockdown mode.

Add Users to the DCUI.Access Advanced Option

If there is a catastrophic failure, the `DCUI.Access` advanced option allows you to exit lockdown mode when you cannot access the host from vCenter Server. You add users to the list by editing the Advanced Settings for the host from the vSphere Client.

Note Users in the DCUI.Access list can change lockdown mode settings regardless of their privileges. The ability to change lockdown modes can impact the security of your host. For service accounts that need direct access to the host, consider adding users to the Exception Users list instead. Exception users can only perform tasks for which they have privileges. See Specify Lockdown Mode Exception Users.

Procedure

1. Browse to the host in the vSphere Client inventory.
2. Click **Configure**.
3. Under System, click **Advanced System Settings**, and click **Edit**.
4. Filter for DCUI.
5. In the **DCUI.Access** text box, enter the local ESXi user names, separated by commas.

   Note You cannot enter Active Directory users. Only local ESXi users are supported.

   By default, the root user is included. Consider removing the root user from the DCUI.Access list, and specifying a named account for better auditability.
6. Click **OK**.

Specify Lockdown Mode Exception Users

You can add users to the Exception Users list from the vSphere Client. These users do not lose their permissions when the host enters lockdown mode. It makes sense to add service accounts such as a backup agent to the Exception Users list.

Exception users do not lose their privileges when the host enters lockdown mode. Usually these accounts represent third-party solutions and external applications that need to continue to function in lockdown mode.

Note The Exception Users list is meant for service accounts that perform very specific tasks, and not for administrators. Adding administrator users to the Exception Users list defeats the purpose of lockdown mode.

Exception users are host local users or Active Directory users with privileges defined locally for the ESXi host. They are not members of an Active Directory group and are not vCenter Server users. These users are allowed to perform operations on the host based on their privileges. That means, for example, that a read-only user cannot disable lockdown mode on a host.

Procedure

1. Browse to the host in the vSphere Client inventory.
2. Click **Configure**.
3. Under System, select **Security Profile**.
4. In the Lockdown Mode panel, click **Edit**.
5. Click **Exception Users** and click the **Add User** icon to add exception users.

Using VIBs to Perform Secure Updates

Upgrading ESXi with ESXCLI requires an understanding of VIBs, image profiles, and software depots.

ESXi consists of an image profile, which describes a set of vSphere Installation Bundles (VIBs) that contain the actual software. A VIB is a signed ramdisk representing a component of the system, roughly analogous to an RPM or DEB on a Linux system. An image profile is a collection of VIBs. A software depot is a collection of VIBs and image profiles. ESXi patches and depots contain updated image profiles composed from a common set of VIBs.

You can install ESXi updates on a standalone host using the `esxcli software` commands. For more information, see the *VMware ESXi Upgrade* documentation.

Note Normally, in a vSphere 7.0 and later environment, you use VMware vSphere® vSphere Lifecycle Manager for lifecycle management of ESXi hosts.

To list all installed VIBs and their current version, or the current image profile, you can use the following ESXCLI commands.

- `esxcli software vib list`
- `esxcli software profile get`

In general, the high-level steps to upgrade ESXi securely are:

- Putting the ESXi host in maintenance mode

- Running an `esxcli software profile update` command, which points to a URL or a ZIP file transferred to the host through SSH
- Restarting the ESXi host

Because VMware cryptographically signs VIBS, secure transfer of VIBs or the entire depot is not necessary, and the update process verifies these signatures.

Manage the Acceptance Levels of Hosts and VIBs

The acceptance level of a VIB depends on the amount of certification of that VIB. The acceptance level of the host depends on the level of the lowest VIB. If you want to allow lower-level VIBs, you can change the acceptance level of the host. You can remove CommunitySupported VIBs to be able to change the host acceptance level.

VIBs are software packages that include a signature from VMware or a VMware partner. To protect the integrity of the ESXi host, do not allow users to install unsigned (community-supported) VIBs. An unsigned VIB contains code that is not certified by, accepted by, or supported by VMware or its partners. Community-supported VIBs do not have a digital signature.

The host's acceptance level must be the same or less restrictive than the acceptance level of any VIB you want to add to the host. For example, if the host acceptance level is VMwareAccepted, you cannot install VIBs at the PartnerSupported level. You can use ESXCLI commands to set an acceptance level for a host. To protect the security and integrity of your ESXi hosts, do not allow unsigned (CommunitySupported) VIBs to be installed on hosts in production systems.

The acceptance level for an ESXi host is displayed in the **Security Profile** in the vSphere Client.

The following acceptance levels are supported.

VMwareCertified

The VMwareCertified acceptance level has the most stringent requirements. VIBs with this level go through thorough testing fully equivalent to VMware in-house Quality Assurance testing for the same technology. Today, only I/O Vendor Program (IOVP) program drivers are published at this level. VMware takes support calls for VIBs with this acceptance level.

VMwareAccepted

VIBs with this acceptance level go through verification testing, but the tests do not fully test every function of the software. The partner runs the tests and VMware verifies the result. Today, CIM providers and PSA plug-ins are among the VIBs published at this level. VMware directs customers with support calls for VIBs with this acceptance level to contact the partner's support organization.

PartnerSupported

VIBs with the PartnerSupported acceptance level are published by a partner that VMware trusts. The partner performs all testing. VMware does not verify the results. This level is used for a new or nonmainstream technology that partners want to enable for VMware systems. Today, driver VIB technologies such as Infiniband, ATAoE, and SSD are at this level with nonstandard hardware drivers. VMware directs customers with support calls for VIBs with this acceptance level to contact the partner's support organization.

CommunitySupported

The CommunitySupported acceptance level is for VIBs created by individuals or companies outside of VMware partner programs. VIBs at this level have not gone through any VMware-approved testing program and are not supported by VMware Technical Support or by a VMware partner.

Procedure

1 Connect to each ESXi host and verify that the acceptance level is set to VMwareCertified, VMwareAccepted, or PartnerSupported by running the following command.

```
esxcli software acceptance get
```

2 If the host acceptance level is CommunitySupported, determine whether any of the VIBs are at the CommunitySupported level by running the following commands.

```
esxcli software vib list
esxcli software vib get -n vibname
```

3 Remove any CommunitySupported VIBs by running the following command.

```
esxcli software vib remove --vibname vib
```

4 Change the acceptance level of the host by using one of the following methods.

Option	Description
CLI command	`esxcli software acceptance set --level level` The `level` parameter is required and specifies the acceptance level to set. Should be one of **`VMwareCertified`**, **`VMwareAccepted`**, **`PartnerSupported`**, or **`CommunitySupported`**. See *ESXCLI Reference* for more information.
vSphere Client	a Select a host in the inventory. b Click **Configure**. c Under System, select **Security Profile**. d Click **Edit** for Host Image Profile Acceptance Level and choose the acceptance level.

Results

The new acceptance level is in effect.

Note ESXi conducts integrity checks of VIBs governed by the Acceptance Level. You can use the `VMkernel.Boot.execInstalledOnly` setting to instruct ESXi to only execute binaries that originate from a valid VIB installed on the host. Combined with Secure Boot, this setting ensures that every single process ever run on an ESXi host is signed, allowed, and expected. By default, the `VMkernel.Boot.execInstalledOnly` setting is disabled for partner compatibility in vSphere 7. Enabling this setting when possible improves security. For more information on configuring advanced options for ESXi, see the VMware knowledge base article at https://kb.vmware.com/kb/1038578.

Assigning Privileges for ESXi Hosts

Usually, you give privileges to users by assigning permissions to ESXi host objects that are managed by a vCenter Server system. If you are using a standalone ESXi host, you can assign privileges directly.

Assigning Permissions to ESXi Hosts That Are Managed by vCenter Server

If your ESXi host is managed by a vCenter Server, perform management tasks through the vSphere Client.

You can select the ESXi host object in the vCenter Server object hierarchy and assign the administrator role to a limited number of users. Those users can then perform direct management on the ESXi host. See Using Roles to Assign Privileges.

Best practice is to create at least one named user account, assign it full administrative privileges on the host, and use this account instead of the root account. Set a highly complex password for the root account and limit the use of the root account. Do not remove the root account.

Assigning Permissions to Standalone ESXi Hosts

You can add local users and define custom roles from the Management tab of the VMware Host Client. See the *vSphere Single Host Management - VMware Host Client* documentation.

For all versions of ESXi, you can see the list of predefined users in the `/etc/passwd` file.

The following roles are predefined.

Read Only

Allows a user to view objects associated with the ESXi host but not to make any changes to objects.

Administrator

Administrator role.

No Access

No access. This role is the default role. You can override the default role.

You can manage local users and groups and add local custom roles to an ESXi host using a VMware Host Client connected directly to the ESXi host. See the *vSphere Single Host Management - VMware Host Client* documentation.

Starting with vSphere 6.0, you can use ESXCLI account management commands for managing ESXi local user accounts. You can use ESXCLI permission management commands for setting or removing permissions on both Active Directory accounts (users and groups) and on ESXi local accounts (users only).

Note If you define a user for the ESXi host by connecting to the host directly, and a user with the same name also exists in vCenter Server, those users are different. If you assign a role to the ESXi user, the vCenter Server user is not assigned the same role.

Predefined Privileges

If your environment does not include a vCenter Server system, the following users are predefined.

root User

By default each ESXi host has a single root user account with the Administrator role. That root user account can be used for local administration and to connect the host to vCenter Server.

Assigning root user privileges can make it easier to break into an ESXi host because the name is already known. Having a common root account also makes it harder to match actions to users.

For better auditing, create individual accounts with Administrator privileges. Set a highly complex password for the root account and limit the use of the root account, for example, for use when adding a host to vCenter Server. Do not remove the root account. For more information about assigning permissions to a user for an ESXi host, see *vSphere Single Host Management - VMware Host Client* documentation.

Best practice is to ensure that any account with the Administrator role on an ESXi host is assigned to a specific user with a named account. Use ESXi Active Directory capabilities, which allow you to manage Active Directory credentials.

Important You can remove the access privileges for the root user. However, you must first create another permission at the root level that has a different user assigned to the Administrator role.

vpxuser User

vCenter Server uses vpxuser privileges when managing activities for the host.

The vCenter Server administrator can perform most of the same tasks on the host as the root user and also schedule tasks, work with templates, and so forth. However, the vCenter Server administrator cannot directly create, delete, or edit local users and groups for hosts. Only a user with Administrator privileges can perform these tasks directly on a host.

Note You cannot manage vpxuser using Active Directory.

Caution Do not change vpxuser in any way. Do not change its password. Do not change its permissions. If you do so, you might experience problems when working with hosts through vCenter Server.

dcui User

The dcui user runs on hosts and acts with Administrator rights. This user's primary purpose is to configure hosts for lockdown mode from the Direct Console User Interface (DCUI).

This user acts as an agent for the direct console and cannot be modified or used by interactive users.

Using Active Directory to Manage ESXi Users

You can configure ESXi to use a directory service such as Active Directory to manage users.

Creating local user accounts on each host presents challenges with having to synchronize account names and passwords across multiple hosts. Join ESXi hosts to an Active Directory domain to eliminate the need to create and maintain local user accounts. Using Active Directory for user authentication simplifies the ESXi host configuration and reduces the risk for configuration issues that could lead to unauthorized access.

When you use Active Directory, users supply their Active Directory credentials and the domain name of the Active Directory server when adding a host to a domain.

Configure a Host to Use Active Directory

You can configure a host to use a directory service such as Active Directory to manage users and groups.

When you add an ESXi host to Active Directory, the DOMAIN group **ESX Admins** is assigned full administrative access to the host if it exists. If you do not want to make full administrative access available, see VMware Knowledge Base article 1025569 for a workaround.

If a host is provisioned with Auto Deploy, Active Directory credentials cannot be stored on the hosts. You can use the vSphere Authentication Proxy to join the host to an Active Directory domain. Because a trust chain exists between the vSphere Authentication Proxy and the host, the Authentication Proxy can join the host to the Active Directory domain. See Using vSphere Authentication Proxy.

Note When you define user account settings in Active Directory, you can limit the computers that a user can log in to by the computer name. By default, no equivalent restrictions are set on a user account. If you set this limitation, LDAP Bind requests for the user account fail with the message `LDAP binding not successful`, even if the request is from a listed computer. You can avoid this problem by adding the netBIOS name for the Active Directory server to the list of computers that the user account can log in to.

Prerequisites

- Verify that you have an Active Directory domain. See your directory server documentation.
- Verify that the host name of ESXi is fully qualified with the domain name of the Active Directory forest.

  fully qualified domain name = host_name.domain_name

Procedure

1. Synchronize the time between ESXi and the directory service system.

   See Synchronize ESXi Clocks with a Network Time Server or the VMware Knowledge Base for information about how to synchronize ESXi time with a Microsoft Domain Controller.
2. Ensure that the DNS servers that you configured for the host can resolve the host names for the Active Directory controllers.
   a. Browse to the host in the vSphere Client inventory.
   b. Click **Configure**.
   c. Under Networking, click **TCP/IP configuration**.
   d. Under TCP/IP Stack: Default, click **DNS** and verify that the host name and DNS server information for the host are correct.

What to do next

Join the host to a directory service domain. See Add a Host to a Directory Service Domain. For hosts that are provisioned with Auto Deploy, set up the vSphere Authentication Proxy. See Using vSphere Authentication Proxy. You can configure permissions so that users and groups from the joined Active Directory domain can access the vCenter Server components. For information about managing permissions, see Add a Permission to an Inventory Object .

Add a Host to a Directory Service Domain

To have your host use a directory service, you must join the host to the directory service domain.

You can enter the domain name in one of two ways:

- **`name.tld`** (for example, **`domain.com`**): The account is created under the default container.
- **`name.tld/container/path`** (for example, **`domain.com/OU1/OU2`**): The account is created under a particular organizational unit (OU).

To use the vSphere Authentication Proxy service, see Using vSphere Authentication Proxy.

Procedure

1. Browse to a host in the vSphere Client inventory.
2. Click **Configure**.
3. Under System, select **Authentication Services**.
4. Click **Join Domain**.
5. Enter a domain.

   Use the form **`name.tld`** or **`name.tld/container/path`**.
6. Enter the user name and password of a directory service user who has permissions to join the host to the domain, and click **OK**.
7. (Optional) If you intend to use an authentication proxy, enter the proxy server IP address.
8. Click **OK** to close the Directory Services Configuration dialog box.

What to do next

You can configure permissions so that users and groups from the joined Active Directory domain can access the vCenter Server components. For information about managing permissions, see Add a Permission to an Inventory Object .

View Directory Service Settings

You can view the type of directory server, if any, that the host uses to authenticate users and the directory server settings.

Procedure

1. Browse to the host in the vSphere Client inventory.
2. Click **Configure**.
3. Under System, select **Authentication Services**.

   The Authentication Services page displays the directory service and domain settings.

What to do next

You can configure permissions so that users and groups from the joined Active Directory domain can access the vCenter Server components. For information about managing permissions, see Add a Permission to an Inventory Object .

Using vSphere Authentication Proxy

You can add ESXi hosts to an Active Directory domain by using vSphere Authentication Proxy instead of adding the hosts explicitly to the Active Directory domain.

You only have to set up the host so it knows about the domain name of the Active Directory server and about the IP address of vSphere Authentication Proxy. When vSphere Authentication Proxy is enabled, it automatically adds hosts that are being provisioned with Auto Deploy to the Active Directory domain. You can also use vSphere Authentication Proxy with hosts that are not provisioned by using Auto Deploy.

See Required Ports for vCenter Server for information about TCP ports used by vSphere Authentication Proxy.

Auto Deploy

If you are provisioning hosts with Auto Deploy, you can set up a reference host that points to Authentication Proxy. You then set up a rule that applies the reference host's profile to any ESXi host that is provisioned with Auto Deploy. vSphere Authentication Proxy stores the IP addresses of all hosts that Auto Deploy provisions using PXE in its access control list. When the host boots, it contacts vSphere Authentication Proxy, and vSphere Authentication Proxy joins those hosts, which are already in its access control list, to the Active Directory domain.

Even if you use vSphere Authentication Proxy in an environment that uses certificates that are provisioned by VMCA or third-party certificates, the process works seamlessly if you follow the instructions for using custom certificates with Auto Deploy.

See Use Custom Certificates with Auto Deploy.

Other ESXi Hosts

You can set up other hosts to use vSphere Authentication Proxy if you want to make it possible for the host to join the domain without using Active Directory credentials. That means you do not need to transmit Active Directory credentials to the host, and you do not save Active Directory credentials in the host profile.

In that case, you add the host's IP address to the vSphere Authentication Proxy access control list, and vSphere Authentication Proxy authorizes the host based on its IP address by default. You can enable client authentication to have vSphere Authentication Proxy check the host's certificate.

Note You cannot use vSphere Authentication Proxy in an environment that supports only IPv6.

Enable vSphere Authentication Proxy

The vSphere Authentication Proxy service is available on each vCenter Server system. By default, the service is not running. If you want to use vSphere Authentication Proxy in your environment, you can start the service from the vCenter Server Management Interface or from the command line.

The vSphere Authentication Proxy service binds to an IPv4 address for communication with vCenter Server, and does not support IPv6. The vCenter Server instance can be on a host machine in an IPv4-only or IPv4/IPv6 mixed-mode network environment. However, when you specify the address of vSphere Authentication Proxy, you must specify an IPv4 address.

Prerequisites

Verify that you are using vCenter Server 6.5 or later. In earlier versions of vSphere, vSphere Authentication Proxy is installed separately. See the documentation for the earlier version of the product for instructions.

Procedure

1 Start the VMware vSphere Authentication Proxy service.

Option	Description
vCenter Server Management Interface	a In a Web browser, go to the vCenter Server Management Interface, https://*vcenter-IP-address-or-FQDN*:5480. b Log in as root. The default root password is the password that you set while deploying the vCenter Server. c Click **Services**, and click the **VMware vSphere Authentication Proxy** service. d Click **Start**. e (Optional) After the service has started, click **Set Startup Type** and click **Automatic** to make the startup automatic.
CLI	`service-control --start vmcam`

2 Confirm that the service started successfully.

Results

You can now set the vSphere Authentication Proxy domain. After that, vSphere Authentication Proxy handles all hosts that are provisioned with Auto Deploy, and you can explicitly add hosts to vSphere Authentication Proxy.

Add a Domain to vSphere Authentication Proxy with the vSphere Client

You can add a domain to vSphere Authentication Proxy from the vSphere Client or by using the `camconfig` command.

You can add a domain to vSphere Authentication Proxy only after you enable the proxy. After you add the domain, vSphere Authentication Proxy adds all hosts that you provision with Auto Deploy to that domain. For other hosts, you can also use vSphere Authentication Proxy if you do not want to give those hosts domain privileges.

Procedure

1 Connect to a vCenter Server system with the vSphere Client.

2 Select the vCenter Server, and click **Configure**.

3 Click **Authentication Proxy** and click **Edit**.

4 Enter the name of the domain that vSphere Authentication Proxy will add hosts to, and the name and password of a user who has Active Directory privileges to add hosts to the domain.

5 Click **Save**.

Add a Domain to vSphere Authentication Proxy with the camconfig Command

You can add a domain to vSphere Authentication using the `camconfig` command.

You can add a domain to vSphere Authentication Proxy only after you enable the proxy. After you add the domain, vSphere Authentication Proxy adds all hosts that you provision with Auto Deploy to that domain. For other hosts, you can also use vSphere Authentication Proxy if you do not want to give those hosts domain privileges.

Procedure

1 Log in to the vCenter Server system as a user with administrator privileges.

2 Run the command to enable access to the Bash shell.

```
shell
```

3 Go to the `/usr/lib/vmware-vmcam/bin/` directory where the **camconfig** script is located.

4 To add the domain and user Active Directory credentials to the Authentication Proxy configuration, run the following command.

```
camconfig add-domain -d domain -u user
```

You are prompted for a password.

vSphere Authentication Proxy caches that user name and password. You can remove and recreate the user as needed. The domain must be reachable through DNS, but does not have to be a vCenter Single Sign-On identity source.

vSphere Authentication Proxy uses the user name specified by *user* to create the accounts for ESXi hosts in Active Directory. The user must have privileges to create accounts in the Active Directory domain to which you are adding the hosts. At the time of writing of this information, the Microsoft Knowledge Base article 932455 had background information for account creation privileges.

5 If you later want to remove the domain and user information from vSphere Authentication Proxy, run the following command.

```
camconfig remove-domain -d domain
```

Use vSphere Authentication Proxy to Add a Host to a Domain

The Auto Deploy server adds all hosts that it provisions to vSphere Authentication Proxy, and vSphere Authentication Proxy adds those hosts to the domain. If you want to add other hosts to a domain using vSphere Authentication Proxy, you can add those hosts to vSphere Authentication Proxy explicitly. Afterwards, the vSphere Authentication Proxy server adds those hosts to the domain. As a result, user-supplied credentials no longer have to be transmitted to the vCenter Server system.

You can enter the domain name in one of two ways:

- **`name.tld`** (for example, **`domain.com`**): The account is created under the default container.
- **`name.tld/container/path`** (for example, **`domain.com/OU1/OU2`**): The account is created under a particular organizational unit (OU).

Prerequisites

- If the ESXi host is using a VMCA-signed certificate, verify that the host has been added to vCenter Server. Otherwise, the Authentication Proxy service cannot trust the ESXi host.
- If the ESXi host is using a root CA-signed certificate, verify that the appropriate root CA-signed certificate has been added to the vCenter Server system. See Certificate Management for ESXi Hosts.

Procedure

1 Browse to the host in the vSphere Client inventory.

2 Click **Configure**.

3 Under **System**, select **Authentication Services**.

4 Click **Join Domain**.

5 Enter a domain.

Use the form **`name.tld`**, for example **`mydomain.com`**, or **`name.tld/container/path`**, for example, **`mydomain.com/organizational_unit1/organizational_unit2`**.

6 Select **Using Proxy Server**.

7 Enter the IP address of the Authentication Proxy server, which is always the same as the IP address of the vCenter Server system.

8 Click **OK**.

Enable Client Authentication for vSphere Authentication Proxy

By default, vSphere Authentication Proxy adds any host if it has the IP address of that host in its access control list. For additional security, you can enable client authentication. If client authentication is enabled, vSphere Authentication Proxy also checks the certificate of the host.

Prerequisites

- Verify that the vCenter Server system trusts the host. By default, when you add a host to vCenter Server, the host is assigned a certificate that is signed by a vCenter Server trusted root CA. vSphere Authentication Proxy trusts vCenter Server trusted root CA.
- If you plan on replacing ESXi certificates in your environment, perform the replacement before you enable vSphere Authentication Proxy. The certificates on the ESXi host must match that of the host's registration.

Procedure

1. Log in to the vCenter Server system as a user with administrator privileges.
2. Run the command to enable access to the Bash shell.

   ```
   shell
   ```

3. Go to the `/usr/lib/vmware-vmcam/bin/` directory where the **camconfig** script is located.
4. Run the following command to enable client authentication.

   ```
   camconfig ssl-cliAuth -e
   ```

   Going forward, vSphere Authentication Proxy checks the certificate of each host that is added.
5. If you later want to disable client authentication again, run the following command.

   ```
   camconfig ssl-cliAuth -n
   ```

Import the vSphere Authentication Proxy Certificate to ESXi Host

By default, ESXi hosts require explicit verification of the vSphere Authentication Proxy certificate. If you are using vSphere Auto Deploy, the Auto Deploy service takes care of adding the certificate to hosts that it provisions. For other hosts, you must add the certificate explicitly.

Prerequisites

- Upload the vSphere Authentication Proxy certificate to a datastore accessible to the ESXi host. Using an SFTP application such WinSCP, you can download the certificate from the vCenter Server host at the following location.

  `/var/lib/vmware/vmcam/ssl/rui.crt`

- Verify that the `UserVars.ActiveDirectoryVerifyCAMCertificate` ESXi advanced setting is set to 1 (the default).

Procedure

1. Select the ESXi host and click **Configure**.
2. Under **System**, select **Authentication Services**.
3. Click **Import Certificate**.
4. Enter the certificate file path following the format `[datastore]/path/certname.crt`, and click **OK**.

Generate a New Certificate for vSphere Authentication Proxy

You can generate a new certificate that is provisioned by VMCA, or a new certificate that includes VMCA as a subordinate certificate.

See Set Up vSphere Authentication Proxy to Use Custom Certificates if you want to use a custom certificate that is signed by a third-party or enterprise CA.

Prerequisites

You must have root or Administrator privileges on the system on which vSphere Authentication Proxy is running.

Procedure

1. Make a copy of `certool.cfg`.

```
cp /usr/lib/vmware-vmca/share/config/certool.cfg /var/lib/vmware/vmcam/ssl/vmcam.cfg
```

2. Edit the copy with some information about your organization, as in the following example.

```
Country = IE
Name = vmcam
Organization = VMware
OrgUnit = vTSU
State = Cork
Locality = Cork
Hostname = test-cam-1.test1.vmware.com
```

3. Generate the new private key in `/var/lib/vmware/vmcam/ssl/`.

```
/usr/lib/vmware-vmca/bin/certool --genkey --privkey=/var/lib/vmware/vmcam/ssl/rui.key --
pubkey=/tmp/vmcam.pub --server=localhost
```

   For *localhost*, supply the FQDN of the vCenter Server.

4 Generate the new certificate in `/var/lib/vmware/vmcam/ssl/` using the key and `vmcam.cfg` file that you created in Step 1 and Step 2.

```
/usr/lib/vmware-vmca/bin/certool --server=localhost --gencert --privkey=/var/lib/
vmware/vmcam/ssl/rui.key --cert=/var/lib/vmware/vmcam/ssl/rui.crt --config=/var/lib/vmware/
vmcam/ssl/vmcam.cfg
```

For *localhost*, supply the FQDN of the vCenter Server.

Set Up vSphere Authentication Proxy to Use Custom Certificates

Using custom certificates with vSphere Authentication Proxy consists of several steps. First you generate a CSR and send it to your CA for signing. Then you place the signed certificate and key file in a location that vSphere Authentication Proxy can access.

By default, vSphere Authentication Proxy generates a CSR during first boot and asks VMCA to sign that CSR. vSphere Authentication Proxy registers with vCenter Server using that certificate. You can use custom certificates in your environment, if you add those certificates to vCenter Server.

Procedure

1 Generate a CSR for vSphere Authentication Proxy.

a Create a configuration file, `/var/lib/vmware/vmcam/ssl/vmcam.cfg`, as in the following example.

```
[ req ]
distinguished_name = req_distinguished_name
encrypt_key = no
prompt = no
string_mask = nombstr
req_extensions = v3_req
[ v3_req ]
basicConstraints = CA:false
keyUsage = nonRepudiation, digitalSignature, keyEncipherment
subjectAltName = DNS:vcenter1.example.com
[ req_distinguished_name ]
countryName = US
stateOrProvinceName = NY
localityName = New York
0.organizationName = Example Inc.
organizationalUnitName = IT Org
commonName = vcenter1.example.com
```

Note the following:

- subjectAltName: Use the format **DNS:***FQDN_of_vCenter_Appliance_to_use_the_CA-signed certificate*.
- commonName: Use the same FQDN of the vCenter Appliance used in subjectAltName.

b Run `openssl` to generate a CSR file and a key file, passing in the configuration file.

```
openssl req -new -nodes -out vmcam.csr -newkey rsa:2048 -keyout /var/lib/vmware/
vmcam/ssl/rui.key -config /var/lib/vmware/vmcam/ssl/vmcam.cfg
```

2 Back up the `rui.crt` certificate and `rui.key` files, which are stored in the following location.

`/var/lib/vmware/vmcam/ssl/rui.crt`

3 Unregister vSphere Authentication Proxy.

a Go to the `/usr/lib/vmware-vmcam/bin` directory where the `camregister` script is located.

b Run the following command.

```
camregister --unregister -a VC_address -u user
```

user must be a vCenter Single Sign-On user that has administrator permissions on vCenter Server.

4 Stop the vSphere Authentication Proxy service.

Tool	Steps
vCenter Server Configuration Management Interface	a In a Web browser, go to the vCenter Server Configuration Management Interface, https://*vcenter-IP-address-or-FQDN*:5480. b Log in as root. The default root password is the password that you set while deploying the vCenter Server. c Click **Services**, and click the **VMware vSphere Authentication Proxy service**. d Click **Stop**.
CLI	`service-control --stop vmcam`

5 Replace the existing `rui.crt` certificate and `rui.key` files with the files that you received from your CA.

6 Restart the vSphere Authentication Proxy service.

7 Reregister vSphere Authentication Proxy explicitly with vCenter Server by using the new certificate and key.

```
camregister --register -a VC_address -u user -c full_path_to_rui.crt -k
full_path_to_rui.key
```

Configuring Smart Card Authentication for ESXi

You can use smart card authentication to log in to the ESXi Direct Console User Interface (DCUI) by using a Personal Identity Verification (PIV), Common Access Card (CAC) or SC650 smart card instead specifying a user name and password.

A smart card is a small plastic card with an embedded integrated circuit chip. Many government agencies and large enterprises use smart card based two-factor authentication to increase the security of their systems and comply with security regulations.

When smart card authentication is enabled on an ESXi host, the DCUI prompts for a smart card and PIN combination instead of the default prompt for a user name and password.

1 When you insert the smart card into the smart card reader, the ESXi host reads the credentials on it.

2 The ESXi DCUI displays your login ID, and prompts for your PIN.

3 After you enter your PIN, the ESXi host matches it with the PIN stored on the smart card and verifies the certificate on the smart card with Active Directory.

4 After successful verification of the smart card certificate, ESXi logs you in to the DCUI.

You can switch to user name and password authentication from the DCUI by pressing F3.

The chip on the smart card locks after a few consecutive incorrect PIN entries, usually three. If a smart card is locked, only selected personnel can unlock it.

Enable Smart Card Authentication

Enable smart card authentication to prompt for smart card and PIN combination to log in to the ESXi DCUI.

Prerequisites

- Set up the infrastructure to handle smart card authentication, such as accounts in the Active Directory domain, smart card readers, and smart cards.
- Configure ESXi to join an Active Directory domain that supports smart card authentication. For more information, see Using Active Directory to Manage ESXi Users .
- Use the vSphere Client to add root certificates. See Certificate Management for ESXi Hosts.

Procedure

1. Browse to the host in the vSphere Client inventory.
2. Click **Configure**.
3. Under System, select **Authentication Services**.

   You see the current smart card authentication status and a list with imported certificates.
4. In the Smart Card Authentication panel, click **Edit**.
5. In the Edit Smart Card Authentication dialog box, select the Certificates page.
6. Add trusted Certificate Authority (CA) certificates, for example, root and intermediary CA certificates.

   Certificates must be in PEM format.
7. Open the Smart Card Authentication page, select the **Enable Smart Card Authentication** check box, and click **OK**.

Disable Smart Card Authentication

Disable smart card authentication to return to the default user name and password authentication for ESXi DCUI login.

Procedure

1. Browse to the host in the vSphere Client inventory.
2. Click **Configure**.
3. Under System, select **Authentication Services**.

   You see the current smart card authentication status and a list with imported certificates.
4. In the Smart Card Authentication panel, click **Edit**.

5 On the Smart Card Authentication page, deselect the **Enable Smart Card Authentication** check box, and click **OK**.

Authenticating With User Name and Password in Case of Connectivity Problems

If the Active Directory (AD) domain server is not reachable, you can log in to the ESXi DCUI by using user name and password authentication to perform emergency actions on the host.

In exceptional circumstances, the AD domain server is not reachable to authenticate the user credentials on the smart card because of connectivity problems, network outage, or disasters. In that case, you can log in to the ESXi DCUI by using the credentials of a local ESXi Administrator user. After logging in, you can perform diagnostics or other emergency actions. The fallback to user name and password login is logged. When the connectivity to AD is restored, smart card authentication is enabled again.

Note Loss of network connectivity to vCenter Server does not affect smart card authentication if the Active Directory (AD) domain server is available.

Using Smart Card Authentication in Lockdown Mode

When enabled, lockdown mode on the ESXi host increases the security of the host and limits access to the DCUI. Lockdown mode might disable the smart card authentication feature.

In normal lockdown mode, only users on the Exception Users list with administrator privileges can access the DCUI. Exception users are host local users or Active Directory users with permissions defined locally for the ESXi host. If you want to use smart card authentication in normal lockdown mode, you must add users to the Exception Users list from the vSphere Client. These users do not lose their permissions when the host enters normal lockdown mode and can log in to the DCUI. For more information, see Specify Lockdown Mode Exception Users.

In strict lockdown mode, the DCUI service is stopped. As a result, you cannot access the host by using smart card authentication.

Using the ESXi Shell

The ESXi Shell is disabled by default on ESXi hosts. You can enable local and remote access to the shell if necessary.

To reduce the risk of unauthorized access, enable the ESXi Shell for troubleshooting only.

The ESXi Shell is independent of lockdown mode. Even if the host is running in lockdown mode, you can still log in to the ESXi Shell if it is enabled.

ESXi Shell

Enable this service to access the ESXi Shell locally.

SSH

Enable this service to access the ESXi Shell remotely by using SSH.

The root user and users with the Administrator role can access the ESXi Shell. Users who are in the Active Directory group ESX Admins are automatically assigned the Administrator role. By default, only the root user can run system commands (such as `vmware -v`) by using the ESXi Shell.

Note Do not enable the ESXi Shell unless you actually need access.

What to read next

- Enable Access to the ESXi Shell

  ESXi Shell and SSH interfaces are disabled by default. Keep these interfaces disabled unless you are performing troubleshooting or support activities. For day-to-day activities, use the vSphere Client, where activity is subject to role-based access control and modern access control methods.

- Use the Direct Console User Interface to Enable Access to the ESXi Shell

  The Direct Console User Interface (DCUI) allows you to interact with the host locally using text-based menus. Evaluate carefully whether the security requirements of your environment support enabling the Direct Console User Interface.

- Log in to the ESXi Shell for Troubleshooting

  Perform ESXi configuration tasks with the vSphere Client, ESXCLI, or VMware PowerCLI. Log in to the ESXi Shell (formerly Tech Support Mode or TSM) for troubleshooting purposes only.

Enable Access to the ESXi Shell

ESXi Shell and SSH interfaces are disabled by default. Keep these interfaces disabled unless you are performing troubleshooting or support activities. For day-to-day activities, use the vSphere Client, where activity is subject to role-based access control and modern access control methods.

Note Access the host by using the vSphere Client, remote command-line tools (ESXCLI and PowerCLI), and published APIs. Do not enable remote access to the host using SSH unless special circumstances require that you enable SSH access.

Prerequisites

If you want to use an authorized SSH key, you can upload it. See ESXi SSH Keys.

Procedure

1 Browse to the host in the inventory.

2 Click **Configure**, then click **Services** under System.

3 Manage ESXi, SSH, or Direct Console UI services.

   a In the Services pane, select the service.

   b Click **Edit Startup Policy** and select the startup policy **Start and stop manually**.

   c To enable the service, click **Start**.

   When you select **Start and stop manually**, the service does not start when you reboot the host. If you want the service to start when you reboot the host, select **Start and stop with host**.

What to do next

Set the availability and idle timeouts for the ESXi Shell. See Create a Timeout for ESXi Shell Availability and Create a Timeout for Idle ESXi Shell Sessions.

Create a Timeout for ESXi Shell Availability

The ESXi Shell is disabled by default. You can set an availability timeout for the ESXi Shell to increase security when you enable the shell.

The availability timeout setting is the amount of time that can elapse before you must log in after the ESXi Shell is enabled. After the timeout period, the service is disabled and users are not allowed to log in.

Procedure

1 Browse to the host in the vSphere Client inventory.

2 Click **Configure**.

3 Under System, select **Advanced System Settings**.

4 Click **Edit**, and select `UserVars.ESXiShellTimeOut`.

5 Enter the idle timeout setting.

   You must restart the SSH service and the ESXi Shell service for the timeout to take effect.

6 Click **OK**.

Results

If you are logged in when the timeout period elapses, your session will persist. However, after you log out or your session is terminated, users are not allowed to log in.

Create a Timeout for Idle ESXi Shell Sessions

If you enable the ESXi Shell on a host, but forget to log out of the session, the idle session remains connected indefinitely. The open connection increases the potential for someone to gain privileged access to the host. Prevent this by setting a timeout for idle sessions.

The idle timeout is the amount of time that can elapse before a user is logged out of an idle interactive session. You can control the amount of time for both local and remote (SSH) session from the Direct Console Interface (DCUI) or from the vSphere Client.

Procedure

1. Browse to the host in the vSphere Client inventory.
2. Click **Configure**.
3. Under System, select **Advanced System Settings**.
4. Click **Edit**, select `UserVars.ESXiShellInteractiveTimeOut`, and enter the timeout setting.

   A value of zero (0) disables the idle time.
5. Restart the ESXi Shell service and the SSH service for the timeout to take effect.

Results

If the session is idle, users are logged out after the timeout period elapses.

Use the Direct Console User Interface to Enable Access to the ESXi Shell

The Direct Console User Interface (DCUI) allows you to interact with the host locally using text-based menus. Evaluate carefully whether the security requirements of your environment support enabling the Direct Console User Interface.

You can use the Direct Console User Interface (DCUI) to enable local and remote access to the ESXi Shell. You access the Direct Console User Interface from the physical console attached to the host. After the host reboots and loads ESXi, press F2 to log in to the DCUI. Enter the credentials that you created when you installed ESXi.

Note Changes made to the host using the Direct Console User Interface, the vSphere Client, ESXCLI, or other administrative tools are committed to permanent storage every hour or upon graceful shutdown. If the host fails before the changes are committed, they might be lost.

Procedure

1. From the Direct Console User Interface, press F2 to access the System Customization menu.
2. Select **Troubleshooting Options** and press Enter.
3. From the Troubleshooting Mode Options menu, select a service to enable.
   - Enable ESXi Shell
   - Enable SSH
4. Press Enter to enable the service.
5. Press Esc until you return to the main menu of the Direct Console User Interface.

What to do next

Set the availability and idle timeouts for the ESXi Shell. See Set Availability Timeout or Idle Timeout for the ESXi Shell.

Set Availability Timeout or Idle Timeout for the ESXi Shell

The ESXi Shell is disabled by default. To increase security when you enable the shell, you can set an availability timeout, an idle timeout, or both.

The two types of timeout apply in different situations.

Idle Timeout

If a user enables the ESXi Shell on a host, but forgets to log out of the session, the idle session remains connected indefinitely. The open connection can increase the potential for someone to gain privileged access to the host. You can prevent this situation by setting a timeout for idle sessions.

Availability Timeout

The availability timeout determines how much time can elapse before you log in after you initially enable the shell. If you wait longer, the service is disabled and you cannot log in to the ESXi Shell.

Prerequisites

Enable the ESXi Shell. See Use the Direct Console User Interface to Enable Access to the ESXi Shell.

Procedure

1. Log in to the ESXi Shell.
2. From the Troubleshooting Mode Options menu, select **Modify ESXi Shell and SSH timeouts** and press Enter.
3. Enter the idle timeout (in seconds) or the availability timeout.

   You must restart the SSH service and the ESXi Shell service for the timeout to take effect.
4. Press Enter and press Esc until you return to the main menu of the Direct Console User Interface.
5. Click **OK**.

Results

- If you set the idle timeout, users are logged out after the session is idle for the specified time.
- If you set the availability timeout, and you do not log in before that timeout elapses, logins become disabled again.

Log in to the ESXi Shell for Troubleshooting

Perform ESXi configuration tasks with the vSphere Client, ESXCLI, or VMware PowerCLI. Log in to the ESXi Shell (formerly Tech Support Mode or TSM) for troubleshooting purposes only.

Procedure

1 Log in to the ESXi Shell using one of the following methods.

- If you have direct access to the host, press Alt+F1 to open the login page on the machine's physical console.
- If you are connecting to the host remotely, use SSH or another remote console connection to start a session on the host.

2 Enter a user name and password recognized by the host.

UEFI Secure Boot for ESXi Hosts

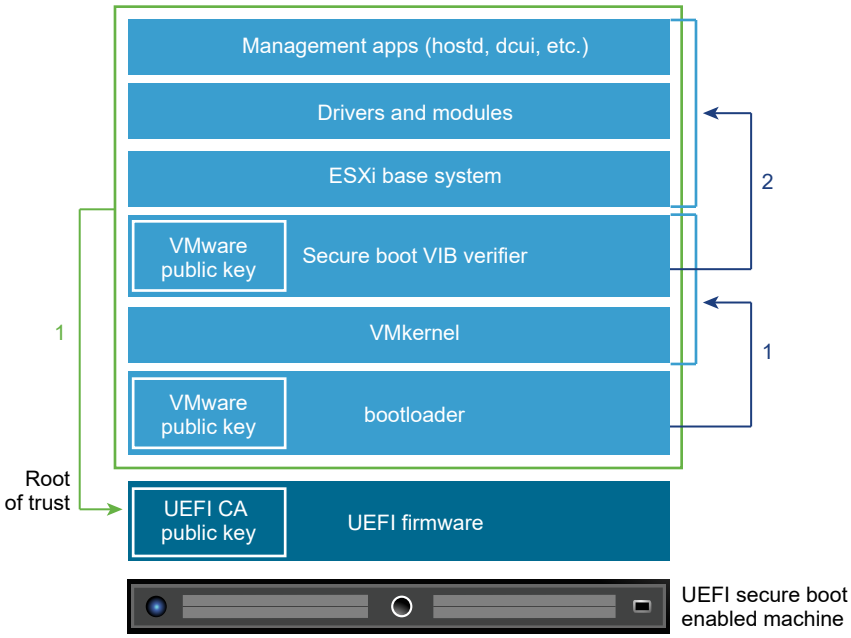
Secure boot is part of the UEFI firmware standard. With secure boot enabled, a machine refuses to load any UEFI driver or app unless the operating system bootloader is cryptographically signed. Starting with vSphere 6.5, ESXi supports secure boot if it is enabled in the hardware.

How ESXi Uses UEFI Secure Boot

ESXi version 6.5 and later supports UEFI Secure Boot at each level of the boot stack.

Note Before you use UEFI Secure Boot on a host that was upgraded, check for compatibility by following the instructions in Run the Secure Boot Validation Script on an Upgraded ESXi Host.

Figure 3-1. UEFI Secure Boot



With secure boot enabled, the boot sequence proceeds as follows.

1 Starting with vSphere 6.5, the ESXi bootloader contains a VMware public key. The bootloader uses this key to verify the signature of the kernel and a small subset of the system that includes a secure boot VIB verifier.

2 The VIB verifier verifies every VIB package that is installed on the system.

At this point, the entire system boots with the root of trust in certificates that are part of the UEFI firmware.

Note When you install or upgrade to vSphere 7.0 Update 2 or later, and an ESXi host has a TPM, the TPM seals the sensitive information by using a TPM policy based on PCR values for UEFI Secure Boot. This value is loaded during subsequent reboots if the policy is satisfied as true. To disable or enable UEFI Secure Boot in vSphere 7.0 Update 2 and later, see Enable or Disable the Secure Boot Enforcement for a Secure ESXi Configuration.

Troubleshooting UEFI Secure Boot

If secure boot does not succeed at any level of the boot sequence, an error results.

The error message depends on the hardware vendor and on the level at which verification did not succeed.

- If you attempt to boot with a bootloader that is unsigned or has been tampered with, an error during the boot sequence results. The exact message depends on the hardware vendor. It might look like the following error, but might look different.

  ```
  UEFI0073: Unable to boot PXE Device...because of the Secure Boot policy
  ```

- If the kernel has been tampered with, an error like the following results.

  ```
  Fatal error: 39 (Secure Boot Failed)
  ```

- If a package (VIB or driver) has been tampered with, a purple screen with the following message appears.

  ```
  UEFI Secure Boot failed:
  Failed to verify signatures of the following vibs (XX)
  ```

To resolve issues with secure boot, follow these steps.

1 Reboot the host with secure boot disabled.

2 Run the secure boot verification script (see Run the Secure Boot Validation Script on an Upgraded ESXi Host).

3 Examine the information in the `/var/log/esxupdate.log` file.

Run the Secure Boot Validation Script on an Upgraded ESXi Host

After you upgrade an ESXi host from an older version of ESXi that did not support UEFI secure boot, you might be able to enable secure boot. Whether you can enable secure boot depends on how you performed the upgrade and whether the upgrade replaced all the existing VIBs or left some VIBs unchanged. You can run a validation script after you perform the upgrade to determine whether the upgraded installation supports secure boot.

For secure boot to succeed, the signature of every installed VIB must be available on the system. Older versions of ESXi do not save the signatures when installing VIBs.

- If you upgrade using ESXCLI commands, the old version of ESXi performs the installation of the new VIBs, so their signatures are not saved and secure boot is not possible.
- If you upgrade using the ISO, new VIBs do have their signatures saved. This is true also for vSphere Lifecycle Manager upgrades that use the ISO.
- If old VIBs remain on the system, the signatures of those VIBs are not available and secure boot is not possible.
  - If the system uses a third-party driver, and the VMware upgrade does not include a new version of the driver VIB, then the old VIB remains on the system after upgrade.
  - In rare cases, VMware might drop ongoing development of a specific VIB without providing a new VIB that replaces or obsoletes it, so the old VIB remains on the system after upgrade.

Note UEFI secure boot also requires an up-to-date bootloader. This script does not check for an up-to-date bootloader.

Prerequisites

- Verify that the hardware supports UEFI secure boot.
- Verify that all VIBs are signed with an acceptance level of at least PartnerSupported. If you include VIBs at the CommunitySupported level, you cannot use secure boot.

Procedure

1. Upgrade the ESXi and run the following command.

   ```
   /usr/lib/vmware/secureboot/bin/secureBoot.py -c
   ```

2. Check the output.

   The output either includes `Secure boot can be enabled` or `Secure boot CANNOT be enabled`.

Securing ESXi Hosts with Trusted Platform Module

ESXi hosts can use Trusted Platform Modules (TPM) chips, which are secure cryptoprocessors that enhance host security by providing a trust assurance rooted in hardware as opposed to software.

TPM is an industry-wide standard for secure cryptoprocessors. TPM chips are found in most of today's computers, from laptops, to desktops, to servers. vSphere 6.7 and later supports TPM version 2.0.

A TPM 2.0 chip attests to an ESXi host's identity. Host attestation is the process of authenticating and attesting to the state of the host's software at a given point in time. UEFI secure boot, which ensures that only signed software is loaded at boot time, is a requirement for successful attestation. The TPM 2.0 chip records and securely stores measurements of the software modules booted in the system, which vCenter Server remotely verifies.

The high-level steps of the remote attestation process are:

1 Establish the trustworthiness of the remote TPM and create an Attestation Key (AK) on it.

   When an ESXi host is added to, rebooted from, or reconnected to vCenter Server, vCenter Server requests an AK from the host. Part of the AK creation process also involves the verification of the TPM hardware itself, to ensure that a known (and trusted) vendor has produced it.

2 Retrieve the Attestation Report from the host.

   vCenter Server requests that the host sends an Attestation Report, which contains a quote of Platform Configuration Registers (PCRs), signed by the TPM, and other signed host binary metadata. By checking that the information corresponds to a configuration it deems trusted, a vCenter Server identifies the platform on a previously untrusted host.

3 Verify the host's authenticity.

   vCenter Server verifies the authenticity of the signed quote, infers the software versions, and determines the trustworthiness of said software versions. If vCenter Server determines the signed quote is invalid, remote attestation fails and the host is not trusted.

To use a TPM 2.0 chip, your vCenter Server environment must meet these requirements:

- vCenter Server 6.7 or later
- ESXi 6.7 host or later with TPM 2.0 chip installed and enabled in UEFI
- UEFI Secure Boot enabled

Ensure that the TPM is configured in the ESXi host's BIOS to use the SHA-256 hashing algorithm and the TIS/FIFO (First-In, First-Out) interface and not CRB (Command Response Buffer). For information about setting these required BIOS options, refer to the vendor documentation.

Review the TPM 2.0 chips certified by VMware at the following location:

https://www.vmware.com/resources/compatibility/search.php

When you boot an ESXi host with an installed TPM 2.0 chip, vCenter Server monitors the host's attestation status. The vSphere Client displays the hardware trust status in the vCenter Server's **Summary** tab under **Security** with the following alarms:

- Green: Normal status, indicating full trust.

- Red: Attestation failed.

Note If you add a TPM 2.0 chip to an ESXi host that vCenter Server already manages, you must first disconnect the host, then reconnect it. See *vCenter Server and Host Management* documentation for information about disconnecting and reconnecting hosts.

 (ESXi and Trusted Platform Module 2.0 Feature Demonstration)

View ESXi Host Attestation Status

When added to an ESXi host, a Trusted Platform Module 2.0 compatible chip attests the integrity of the platform. You can view the attestation status of the host in the vSphere Client. You can also view the Intel Trusted Execution Technology (TXT) status.

Procedure

1. Connect to vCenter Server by using the vSphere Client.
2. Navigate to a data center and click the **Monitor** tab.
3. Click **Security**.
4. Review the host's status in the Attestation column and read the accompanying message in the **Message** column.
5. If this host is a Trusted Host, see View the Trusted Cluster Attestation Status for more information.

What to do next

For a Failed or Warning attestation status, see Troubleshoot ESXi Host Attestation Problems. For Trusted Hosts, see Troubleshoot Trusted Host Attestation Problems.

Troubleshoot ESXi Host Attestation Problems

When you install a Trusted Platform Module (TPM) device on an ESXi host, the host might fail to pass attestation. You can troubleshoot the potential causes of this problem.

Procedure

1. View the ESXi host alarm status and accompanying error message. See View ESXi Host Attestation Status.
2. If the error message is `Host secure boot was disabled`, you must re-enable secure boot to resolve the problem.
3. If the attestation status of the host is failed, check the vCenter Server `vpxd.log` file for the following message:

   ```
   No cached identity key, loading from DB
   ```

This message indicates that you are adding a TPM 2.0 chip to an ESXi host that vCenter Server already manages. You must first disconnect the host, then reconnect it. See *vCenter Server and Host Management* documentation for information about disconnecting and reconnecting hosts.

For more information about vCenter Server log files, including location and log rotation, see the VMware knowledge base article at https://kb.vmware.com/s/article/1021804.

4 For all other error messages, contact Customer Support.

ESXi Log Files

Log files are an important component of troubleshooting attacks and obtaining information about breaches. Logging to a secure, centralized log server can help prevent log tampering. Remote logging also provides a long-term audit record.

To increase the security of the host, take the following measures.

- Configure persistent logging to a datastore. By default, the logs on ESXi hosts are stored in the in-memory file system. Therefore, they are lost when you reboot the host, and only 24 hours of log data is stored. When you enable persistent logging, you have a dedicated activity record for the host.
- Remote logging to a central host allows you to gather log files on a central host. From that host, you can monitor all hosts with a single tool, do aggregate analysis, and search log data. This approach facilitates monitoring and reveals information about coordinated attacks on multiple hosts.
- Configure the remote secure syslog on ESXi hosts by using ESXCLI or PowerCLI, or by using an API client.
- Query the syslog configuration to make sure that the syslog server and port are valid.

See the *vSphere Monitoring and Performance* documentation for information about syslog setup, and for additional information on ESXi log files.

Configure Syslog on ESXi Hosts

You can use the vSphere Client or the `esxcli system syslog` command to configure the syslog service.

For information about using the `esxcli system syslog` command and other ESXCLI commands, see *Getting Started with ESXCLI*.

Procedure

1 Browse to the host in the vSphere Client inventory.

2 Click **Configure**.

3 Under System, click **Advanced System Settings**.

4 Click **Edit**.

5 Filter for **`syslog`**.

6 To set up logging globally, select the setting to change and enter the value.

Option	Description
Syslog.global.defaultRotate	Maximum number of archives to keep. You can set this number globally and for individual subloggers.
Syslog.global.defaultSize	Default size of the log, in KB, before the system rotates logs. You can set this number globally and for individual subloggers.
Syslog.global.LogDir	Directory where logs are stored. The directory can be on mounted NFS or VMFS volumes. Only the `/scratch` directory on the local file system is persistent across reboots. Specify the directory as [*datastorename*] *path_to_file*, where the path is relative to the root of the volume backing the datastore. For example, the path `[storage1] /systemlogs` maps to the path `/vmfs/volumes/storage1/systemlogs`.
Syslog.global.logDirUnique	Selecting this option creates a subdirectory with the name of the ESXi host under the directory specified by **Syslog.global.LogDir**. A unique directory is useful if the same NFS directory is used by multiple ESXi hosts.
Syslog.global.LogHost	Remote host to which syslog messages are forwarded and port on which the remote host receives syslog messages. You can include the protocol and the port, for example, `ssl://hostName1:1514`. UDP (only on port 514), TCP, and SSL are supported. The remote host must have syslog installed and correctly configured to receive the forwarded syslog messages. For more information about remote host configuration, see the documentation for the syslog service installed on the remote host. You can use an unlimited number of remote hosts to receive syslog messages.

7 (Optional) To overwrite the default log size and log rotation for any of the logs:

a Click the name of the log that you want to customize.

b Enter the number of rotations and the log size you want.

8 Click **OK**.

Results

Changes to the syslog options take effect immediately.

ESXi Log File Locations

ESXi records host activity in log files, using a syslog facility.

Table 3-8. ESXi Log File Locations

Component	Location	Purpose
Authentication	`/var/log/auth.log`	Contains all events related to authentication for the local system.
ESXi host agent log	`/var/log/hostd.log`	Contains information about the agent that manages and configures the ESXi host and its virtual machines.
Shell log	`/var/log/shell.log`	Contains a record of all commands typed into the ESXi Shell and shell events (for example, when the shell was enabled).
System messages	`/var/log/syslog.log`	Contains all general log messages and can be used for troubleshooting. This information was formerly located in the messages log file.
vCenter Server agent log	`/var/log/vpxa.log`	Contains information about the agent that communicates with vCenter Server (if the host is managed by vCenter Server).
Virtual machines	The same directory as the affected virtual machine's configuration files, named `vmware.log` and `vmware*.log`. For example, `/vmfs/volumes/datastore/virtual machine/vwmare.log`	Contains virtual machine power events, system failure information, tools status and activity, time sync, virtual hardware changes, vMotion migrations, machine clones, and so on.
VMkernel	`/var/log/vmkernel.log`	Records activities related to virtual machines and ESXi.
VMkernel summary	`/var/log/vmksummary.log`	Used to determine uptime and availability statistics for ESXi (comma separated).
VMkernel warnings	`/var/log/vmkwarning.log`	Records activities related to virtual machines.
Quick Boot	`/var/log/loadESX.log`	Contains all events related to restarting an ESXi host through Quick Boot.
Trusted infrastructure agent	`/var/run/log/kmxa.log`	Records activities related to the Client Service on the ESXi Trusted Host.
Key Provider Service	`/var/run/log/kmxd.log`	Records activities related to the vSphere Trust Authority Key Provider Service.
Attestation Service	`/var/run/log/attestd.log`	Records activities related to the vSphere Trust Authority Attestation Service.

Table 3-8. ESXi Log File Locations (continued)

Component	Location	Purpose
ESX Token Service	`/var/run/log/esxtokend.log`	Records activities related to the vSphere Trust Authority ESX Token Service.
ESX API Forwarder	`/var/run/log/esxapiadapter.log`	Records activities related to the vSphere Trust Authority API forwarder.

Securing Fault Tolerance Logging Traffic

VMware Fault Tolerance (FT) captures inputs and events that occur on a primary VM and sends them to the secondary VM, which is running on another host.

This logging traffic between the primary and secondary VMs is unencrypted and contains guest network and storage I/O data, as well as the memory contents of the guest operating system. This traffic might include sensitive data such as passwords in plaintext. To avoid such data being divulged, ensure that this network is secured, especially to avoid man-in-the-middle attacks. For example, use a private network for FT logging traffic.

Enable Fault Tolerance Encryption

You can encrypt Fault Tolerance log traffic.

vSphere Fault Tolerance performs frequent checks between a primary VM and secondary VM so that the secondary VM can quickly resume from the last successful checkpoint. The checkpoint contains the VM state that has been modified since the previous checkpoint. You can encrypt Fault Tolerance log traffic.

When you turn on Fault Tolerance, FT encryption is set to **Opportunistic** by default, which means it enables encryption only if both the primary and secondary host are capable of encryption. Follow this procedure if you need to change the FT encryption mode manually.

Note Fault Tolerance supports vSphere Virtual Machine Encryption with vSphere 7.0 Update 2 and later. In-guest and array-based encryption do not depend on or interfere with VM encryption. Having multiple encryption layers uses additional compute resources, which might impact virtual machine performance. The impact varies with hardware as well as the amount and type of I/O, but overall performance impact is negligible for most workloads. The effectiveness and compatibility of back-end storage features such as deduplication, compression, and replication might also be affected by VM encryption.

Prerequisites

FT encryption requires SMP-FT. Encryption on Legacy FT (Record-Replay FT) is not supported.

Procedure

1 Select the VM and choose **Edit Settings**.

2 Under **VM Options** select the **Encrypted FT** drop-down menu.

3 Choose one of the following options:

Option	Description
Disabled	Do not turn on encrypted Fault Tolerance logging.
Opportunistic	Turn on encryption only if both sides are capable. A Fault Tolerance VM is allowed to move to an ESXi host which does not support encrypted Fault Tolerance logging.
Required	Choose hosts for Fault Tolerance primary and secondary that both support encrypted FT logging.

Note While VM encryption is enabled, FT encryption mode is set to **Required** by default and cannot be modified.

When FT encryption mode is set to **Required**:

- When you turn on FT, only FT encryption supported hosts are listed for the placement of FT secondary.
- FT failover can only happen on the FT encryption supported hosts.

4 Click **OK**.

Managing ESXi Audit Records

Audit records conform to RFC 5424 and contain information about events pertaining to items such as the time, status, description, and user information logged for events that have occurred from actions on ESXi hosts. Both local and remote audit record keeping are available. Audit record keeping is deactivated by default. You must manually activate both local and remote auditing modes.

The local ESXi audit log operates as a fixed-size buffer of recent audit messages. Once messages fill the buffer, new records overwrite the oldest records. The remote audit log forwards the same stream of audit records in a standard syslog format (RFC 3164) to a remote server, either in unencrypted or encrypted (RFC 5425) form. Audit messages comply with RFC 5424 but general syslog messages comply only with RFC 3164. The system sends generated audit message to the local store and the remote store simultaneously.

During a loss of connection between the host and the remote store, the remote store drops any generated audit messages. Upon reconnection, the system generates an audit message indicating potential message loss.

Configuring Audit Records

You use ESXCLI to configure the local audit record keeping. For more information, see *ESXCLI Reference* at https://code.vmware.com/.

Viewing Audit Records

You can view the audit records as follows.

- Local: Use the ESXi `/bin/viewAudit` application.
- Remote: Configure a remote audit server using ESXCLI. For more information, see Enable the Transmission of Audit Records to a Remote Host with ESXCLI.

You can also use the FetchAuditRecords API (in the DiagnosticsManager managed object) to view audit records.

Securing the ESXi Configuration

Starting in vSphere 7.0 Update 2, the ESXi configuration is protected by encryption. When an ESXi host is optionally protected by a TPM, the ESXi configuration encryption key is sealed by the TPM.

Many ESXi services store secrets in their configuration files. These configurations persist in an ESXi host's boot bank as an archived file. Beginning in vSphere 7.0 Update 2, this archived file is encrypted. As a result, attackers cannot read or alter this file directly, even if they have physical access to the ESXi host's storage.

In addition to preventing an attacker from accessing secrets, a secure ESXi configuration when used with a TPM can save virtual machine encryption keys across reboots. As a result, encrypted workloads can continue to function when a key server is unavailable or unreachable. See Key Persistence Overview.

Securing the ESXi Configuration Overview

You do not need to enable ESXi configuration encryption manually. When you install or upgrade to vSphere 7.0 Update 2 or later, the archived ESXi configuration file is encrypted.

Before vSphere 7.0 Update 2, the archived ESXi configuration file is not encrypted. In vSphere 7.0 Update 2 and later, the archived configuration file is encrypted. When the ESXi host is configured with a Trusted Platform Module (TPM), the TPM is used to "seal" the configuration to the host, providing a strong security guarantee.

ESXi Configuration Files Overview before vSphere 7.0 Update 2

The configuration of an ESXi host consists of configuration files for each service that runs on the host. The configuration files typically reside in the `/etc/` directory, but they can also reside in other namespaces. The configuration files contain run-time information about the state of the services. Over time, the default values in the configuration files can change, for example, when you change settings on the ESXi host. A cron job backs up the ESXi configuration files periodically, or when ESXi shuts down gracefully, or on demand, and creates an archived

configuration file in the boot bank. When ESXi reboots, it reads the archived configuration file and recreates the state that ESXi was in when the backup was taken. Before vSphere 7.0 Update 2, the archived configuration file is unencrypted. As a result, it is possible for an attacker who has access to the physical ESXi storage to read and alter this file while the system is offline.

Secure ESXi Configuration Overview

During the first boot after installing or upgrading the ESXi host to vSphere 7.0 Update 2 or later, the following occurs:

- If the ESXi host has a TPM, and it is enabled in the firmware, the archived configuration file is encrypted by an encryption key stored in the TPM. From this point on, the configuration of the host is sealed by the TPM.
- If the ESXi host does not have a TPM, ESXi uses a Key Derivation Function (KDF) to generate a secure configuration encryption key for the archived configuration file. The inputs to the KDF are stored on disk in the `encryption.info` file.

Note When an ESXi host has an enabled TPM device, you gain additional protection.

When the ESXi host reboots after the first boot, the following occurs:

- If the ESXi host has a TPM, the host must obtain the encryption key from the TPM for that specific host. If the TPM measurements satisfy the sealing policy that was used when creating the encryption key, then the host obtains the encryption key from the TPM.
- If the ESXi host does not have a TPM, ESXi reads information from the `encryption.info` file to unlock the secure configuration.

Secure ESXi Configuration Requirements

- ESXi 7.0 Update 2 or later
- TPM 2.0 for configuration encryption and ability to use a sealing policy

Secure ESXi Configuration Recovery Key

A secure ESXi configuration includes a recovery key. If you must recover the ESXi secure configuration, you use a recovery key whose contents you enter as a command-line boot option. You can list the recovery key to create a recovery key backup. You can also rotate the recovery key as part of your security requirements.

Taking a backup of the recovery key is an important part of managing your secure ESXi configuration. vCenter Server generates an alarm to remind you to back up the recovery key.

Recovery Key Alarm

Taking a backup of the recovery key is an important part of managing your secure ESXi configuration. Whenever an ESXi host in TPM mode is connected or reconnected to vCenter Server, vCenter Server generates an alarm to remind you to back up the recovery key. When you reset the alarm, it is not triggered again unless conditions change.

Best Practices for Secure ESXi Configuration

Follow these best practices for the recovery key:

- When you list a recovery key, it is temporarily displayed in an untrusted environment and is in memory. Remove traces of the key.
  - Rebooting the host removes the residual key in memory.
  - For enhanced protection, you can enable encryption mode on the host. See Enable Host Encryption Mode Explicitly.
- When you perform a recovery:
  - To eliminate any traces of the recovery key in an untrusted environment, reboot the host.
  - For enhanced security, rotate the recovery key to use a new key after having recovered the key one time.

TPM Sealing Policies Overview

In vSphere 7.0 Update 2 and later, an ESXi host uses the TPM to seal the host's configuration against a Platform Configuration Register (PCR) policy. The PCR policy can be configured to enforce UEFI Secure Boot and other settings.

A TPM can use Platform Configuration Register (PCR) measurements to implement policies that restrict unauthorized access to sensitive data. When you install or upgrade an ESXi host with a TPM to vSphere 7.0 Update 2 and later, the TPM seals the sensitive information by using a policy that incorporates the secure boot setting. This policy checks that if secure boot was enabled when data was first sealed with the TPM, then secure boot must still be enabled when attempting to unseal the data on a subsequent boot.

Secure boot is part of the UEFI firmware standard. With UEFI Secure Boot enabled, a host refuses to load any UEFI driver or app unless the operating system bootloader has a valid digital signature.

You can choose to disable or enable UEFI Secure Boot enforcement. See Enable or Disable the Secure Boot Enforcement for a Secure ESXi Configuration.

Note If you do not activate a TPM when you install or upgrade to vSphere 7.0 Update 2 or later, you can do so later with the following command.

```
esxcli system settings encryption set --mode=TPM
```

Once you have activated the TPM, you cannot undo the setting.

The `esxcli system settings encryption set` command fails on some TPMs even when the TPM is enabled for the host.

- In vSphere 7.0 Update 2: TPMs from NationZ (NTZ), Infineon Technologies (IFX), and certain new models (like NPCT75x) from Nuvoton Technologies Corporation (NTC)
- In vSphere 7.0 Update 3: TPMs from NationZ (NTZ)

If an installation or upgrade of vSphere 7.0 Update 2 or later is unable to use the TPM during the first boot, the installation or upgrade continues, and the mode defaults to NONE (that is, `--mode=NONE`). The resulting behavior is as though the TPM is not activated.

The TPM can also enforce the setting for the execInstalledOnly boot option in the sealing policy. The execInstalledOnly enforcement is an advanced ESXi boot option that guarantees that the VMkernel executes only binaries that have been properly packaged and signed as part of a VIB. The execInstalledOnly boot option has a dependency on the secure boot option. The secure boot enforcement must be enabled before you can enforce the execInstalledOnly boot option in the sealing policy. See Enable or Disable the execInstalledOnly Enforcement for a Secure ESXi Configuration.

Managing a Secure ESXi Configuration

You can use ESXCLI commands to list the secure ESXi configuration recovery key, rotate the recovery key, and change the TPM policies (for example, enforcing UEFI Secure Boot).

List the Contents of the Secure ESXi Configuration Recovery Key

You can use ESXCLI to show the contents of the secure ESXi configuration recovery key.

This task applies only to an ESXi host that has a TPM. In general, you list the contents of the secure ESXi configuration recovery key to create a backup, or as part of rotating recovery keys.

Prerequisites

- Have access to the ESXCLI command set. You can run ESXCLI commands remotely, or run them in the ESXi Shell.
- Required privilege for using ESXCLI standalone version or through PowerCLI: **Host.Config.Settings**

Procedure

1 Run the following command on the ESXi host.

```
esxcli system settings encryption recovery list
```

2 Save the output in a secure, remote location as a backup, in case you must recover the secure configuration.

Results

The recovery key ID and key are displayed.

Example: List the Secure ESXi Configuration Recovery Key

```
[root@host1] esxcli system settings encryption recovery list

Recovery ID                             Key
--------------------------------------  ---
{2DDD5424-7F3F-406A-8DA8-D62630F6C8BC}
478269-039194-473926-430939-686855-231401-642208-184477-602511
-225586-551660-586542-338394-092578-687140-267425
```

Rotate the Secure ESXi Configuration Recovery Key

You can use ESXCLI to rotate the secure ESXi configuration recovery key.

This task applies only to an ESXi host that has a TPM. You can rotate the ESXi secure configuration recovery key as part of your security best practices.

Prerequisites

- Have access to the ESXCLI command set. You can run ESXCLI commands remotely, or run them in the ESXi Shell.
- Required privilege for using ESXCLI standalone version or through PowerCLI: **Host.Config.Settings**

Procedure

1 List the recovery key.

See List the Contents of the Secure ESXi Configuration Recovery Key.

2 Run the following command.

```
esxcli system settings encryption recovery rotate [-k keyID] -u uuid
```

In this command, the optional *keyID* is the key ID in the VMkernel key cache and *uuid* is the Recovery ID (obtained from the `esxcli system settings encryption recovery list` command). If you do not supply the optional key ID, ESXi replaces the old recovery key with a new recovery key that is randomly generated.

Results

The recovery key is now set to the contents of the key referenced by key ID, if provided. Otherwise, ESXi provides a new key ID.

Troubleshooting and Recovering the Secure ESXi Configuration

You can troubleshoot and recover from boot problems that you might encounter with a secure ESXi Configuration.

If you clear a TPM (that is, the seed values in the TPM are reset), if a TPM fails, or if you replace the motherboard or TPM device, or both, you must take steps to recover the ESXi secure configuration. You must have the recovery key to recover the configuration. Until you recover the configuration, the ESXi host cannot boot. See Recover the Secure ESXi Configuration.

Although uncommon, it is possible that an ESXi host might fail to restore or decrypt the secure configuration, preventing the host from booting. Possible situations include:

- Change to secure boot setting (or other policy)
- Actual tampering
- The recovery key is unavailable

To troubleshoot these conditions, see the VMware knowledge base article at https://kb.vmware.com/kb/81446.

Recover the Secure ESXi Configuration

If a TPM fails, or if you clear a TPM, you must recover the secure ESXi Configuration. Until you recover the configuration, the ESXi host cannot boot.

Recovering the secure ESXi configuration refers to the following situations:

- You cleared the TPM (that is, the seeds in the TPM were reset).
- The TPM failed.
- You replaced the motherboard or the TPM device, or both.

To troubleshoot other secure ESXi configuration problems, see the VMware knowledge base article at https://kb.vmware.com/kb/81446.

Perform the recovery manually. Do not perform the recovery as part of an installation or upgrade script.

Prerequisites

Get your recovery key. You should have previously listed and stored the recover key. See List the Contents of the Secure ESXi Configuration Recovery Key.

Procedure

1. (Optional) If the TPM failed, move the disk (having the boot bank) to another host with a TPM.
2. Start the ESXi host.

3 When the ESXi installer window appears, press Shift+O to edit boot options.

4 To recover the configuration, at the command prompt, append the following boot option to any existing boot options.

```
encryptionRecoveryKey=recovery_key
```

The secure ESXi configuration is recovered and the ESXi host boots.

5 To persist the change, enter the following command:

```
/sbin/auto-backup.sh
```

What to do next

When you enter the recovery key, it is temporarily displayed in an untrusted environment and is in memory. Though not necessary, as a best practice, you can remove residual traces of the key in memory by rebooting the host. Or, you can rotate the key. See Rotate the Secure ESXi Configuration Recovery Key.

Enable or Disable the Secure Boot Enforcement for a Secure ESXi Configuration

You can choose to enable UEFI secure boot enforcement, or disable a previously enabled UEFI secure boot enforcement. You must use ESXCLI to change the setting in the TPM on the ESXi host.

This task applies only to ESXi hosts that have a TPM. UEFI Secure boot is a firmware setting for ensuring that the software launched by the firmware is trusted. To learn more, see UEFI Secure Boot for ESXi Hosts. The enablement of UEFI Secure boot can be enforced upon every boot by using the TPM.

Prerequisites

- Have access to the ESXCLI command set. You can run ESXCLI commands remotely, or run them in the ESXi Shell.
- Required privilege for using ESXCLI standalone version or through PowerCLI: **Host.Config.Settings**

Procedure

1 List the current settings on the ESXi host.

```
esxcli system settings encryption get
   Mode: TPM
   Require Executables Only From Installed VIBs: false
   Require Secure Boot: true
```

If secure boot enforcement is enabled, Require Secure Boot displays true. If secure boot enforcement is disabled, Require Secure Boot displays false.

If Mode appears as NONE, you must enable the TPM in the host's firmware and set the mode by running the following command:

```
esxcli system settings encryption set --mode=TPM
```

2 Enable or disable the secure boot enforcement.

Option	Description

Enable	a Shut down the host gracefully.
	For example, right-click the ESXi host in the vSphere Client and select **Power > Shut Down**.
	b Enable secure boot in the firmware of the host.
	See your specific vendor hardware documentation.
	c Restart the host.
	d Run the following ESXCLI command.
	`esxcli system settings encryption set --require-secure-boot=T`
	e Verify the change.
	`esxcli system settings encryption get`
	`Mode: TPM`
	`Require Executables Only From Installed VIBs: false`
	`Require Secure Boot: true`
	Confirm that Required Secure Boot displays true.
	f To save the setting, run the following command.
	`/sbin/auto-backup.sh`

Disable	a Run the following ESXCLI command.
	`esxcli system settings encryption set --require-secure-boot=F`
	b Verify the change.
	`esxcli system settings encryption get`
	`Mode: TPM`
	`Require Executables Only From Installed VIBs: false`
	`Require Secure Boot: false`
	Confirm that Require Secure Boot displays false.
	c To save the setting, run the following command.
	`/sbin/auto-backup.sh`
	You can choose to disable the secure boot in the firmware of the host, but at this point the dependency between the firmware setting and the TPM enforcement is no longer set.

Results

The ESXi host runs with secure boot enforcement enabled or disabled, depending on your choice.

Note If you do not activate a TPM when you install or upgrade to vSphere 7.0 Update 2 or later, you can do so later with the following command.

```
esxcli system settings encryption set --mode=TPM
```

Once you have activated the TPM, you cannot undo the setting.

The `esxcli system settings encryption set` command fails on some TPMs even when the TPM is enabled for the host.

- In vSphere 7.0 Update 2: TPMs from NationZ (NTZ), Infineon Technologies (IFX), and certain new models (like NPCT75x) from Nuvoton Technologies Corporation (NTC)
- In vSphere 7.0 Update 3: TPMs from NationZ (NTZ)

If an installation or upgrade of vSphere 7.0 Update 2 or later is unable to use the TPM during the first boot, the installation or upgrade continues, and the mode defaults to NONE (that is, `--mode=NONE`). The resulting behavior is as though the TPM is not activated.

Enable or Disable the execInstalledOnly Enforcement for a Secure ESXi Configuration

You can choose to enable execInstalledOnly enforcement, or disable a previously enabled execInstalledOnly enforcement. You must use ESXCLI to change the setting in the TPM on the ESXi host. UEFI secure boot enforcement must be enabled before you can enable the execInstalledOnly enforcement.

This task applies only to ESXi hosts that have a TPM. The execInstalledOnly advanced ESXi boot option, when set to TRUE, guarantees that the VMkernel executes only those binaries that have been packaged and signed as part of a VIB. The enablement of this boot option can be enforced upon every boot by using the TPM.

Prerequisites

- To enable the execInstalledOnly enforcement, you must first enable the UEFI secure boot enforcement. The execInstalledOnly enforcement is built on top of the UEFI secure boot enforcement. See Enable or Disable the Secure Boot Enforcement for a Secure ESXi Configuration.
- Have access to the ESXCLI command set. You can run ESXCLI commands remotely, or run them in the ESXi Shell.
- Required privilege for using ESXCLI standalone version or through PowerCLI: **Host.Config.Settings**

Procedure

1 List the current settings on the ESXi host.

```
esxcli system settings encryption get
   Mode: TPM
   Require Executables Only From Installed VIBs: false
   Require Secure Boot: true
```

If execInstalledOnly enforcement is enabled, Require Executables Only From Installed VIBs displays true. If execInstalledOnly enforcement is disabled, Require Executables Only From Installed VIBs displays false. To enable the execInstalledOnly enforcement, the secure boot enforcement must be enabled, and Require Secure Boot displays true in this case.

If Mode appears as NONE, you must enable the TPM in the host's firmware and set the mode by running the following command:

```
esxcli system settings encryption set --mode=TPM
```

Also, if Require Secure Boot displays as False, see Enable or Disable the Secure Boot Enforcement for a Secure ESXi Configuration to enable the enforcement.

2 Enable or disable the execInstalledOnly enforcement.

Option	Description

Enable	a Verify that the secure boot option is enforced.
	`esxcli system settings encryption get`
	`Mode: TPM`
	`Require Executables Only From Installed VIBs: false`
	`Require Secure Boot: true`
	Confirm that Require Secure Boot displays true. If not, see Enable or Disable the Secure Boot Enforcement for a Secure ESXi Configuration.
	b To configure the runtime value of the execInstalledOnly boot option to TRUE, run the following ESXCLI command.
	`esxcli system settings kernel set -s execInstalledOnly -v TRUE`
	c Shut down the host gracefully.
	For example, right-click the ESXi host in the vSphere Client and select **Power > Shut Down**.
	d Restart the host.
	e To set the execInstalledOnly enforcement, run the following ESXCLI command.
	`esxcli system settings encryption set --require-exec-installed-only=T`
	f Verify the change.
	`esxcli system settings encryption get`
	`Mode: TPM`
	`Require Executables Only From Installed VIBs: true`
	`Require Secure Boot: true`
	Confirm that Require Executables Only From Installed VIBs displays true.
	g To save the setting, run the following command.
	`/sbin/auto-backup.sh`

Disable	a Run the following ESXCLI command.
	`esxcli system settings encryption set --require-exec-installed-only=F`
	b Verify the change.
	`esxcli system settings encryption get`
	`Mode: TPM`
	`Require Executables Only From Installed VIBs: false`
	`Require Secure Boot: true`
	Confirm that Require Executables Only From Installed VIBs displays false.
	c To save the setting, run the following command.
	`/sbin/auto-backup.sh`

Option	Description
	The TPM no longer enforces the execInstalledOnly boot option.

Results

The ESXi host runs with execInstalledOnly enforcement enabled or disabled, depending on your choice.

Securing vCenter Server Systems 4

Securing vCenter Server includes ensuring security of the host where vCenter Server is running, following best practices for assigning privileges and roles, and verifying the integrity of the clients that connect to vCenter Server.

Read the following topics next:

- vCenter Server Security Best Practices
- Verify Thumbprints for Legacy ESXi Hosts
- Required Ports for vCenter Server

vCenter Server Security Best Practices

Following vCenter Server security best practices helps you ensure the integrity of your vSphere environment.

Best Practices for vCenter Server Access Control

Strictly control access to different vCenter Server components to increase security for the system.

The following guidelines help ensure security of your environment.

Use Named Accounts

- Grant the Administrator role only to those administrators who are required to have it. You can create custom roles or use the No cryptography administrator role for administrators with more limited privileges. Do not apply this role to any group whose membership is not strictly controlled.
- Make sure that applications use unique service accounts when connecting to a vCenter Server system.

Monitor Privileges of vCenter Server Administrator Users

Not all administrator users must have the Administrator role. Instead, create a custom role with the appropriate set of privileges and assign it to other administrators.

Users with the vCenter Server Administrator role have privileges on all objects in the hierarchy. For example, by default the Administrator role allows users to interact with files and programs inside a virtual machine's guest operating system. Assigning that role to too many users can lessen virtual machine data confidentiality, availability, or integrity. Create a role that gives the administrators the privileges they need, but remove some of the virtual machine management privileges.

Minimize Access

Do not allow users to log directly in to the vCenter Server host machine. Users who are logged in to the vCenter Server host machine can cause harm, either intentionally or unintentionally, by altering settings and modifying processes. Those users also have potential access to vCenter credentials, such as the SSL certificate. Allow only users who have legitimate tasks to perform to log in to the system and ensure that login events are audited.

Grant Minimal Privileges to vCenter Server Database Users

The database user requires only certain privileges specific to database access.

Some privileges are required only for installation and upgrade. You can remove these privileges from the database administrator after vCenter Server is installed or upgraded.

Restrict Datastore Browser Access

Assign the **Datastore.Browse datastore** privilege only to users or groups who really need those privileges. Users with the privilege can view, upload, or download files on datastores associated with the vSphere deployment through the Web browser or the vSphere Client.

Restrict Users From Running Commands in a Virtual Machine

By default, a user with the vCenter Server Administrator role can interact with files and programs within a virtual machine's guest operating system. To reduce the risk of breaching guest confidentiality, availability, or integrity, create a custom nonguest access role without the **Guest Operations** privilege. See Restrict Users from Running Commands Within a Virtual Machine.

Consider Modifying the Password Policy for vpxuser

By default, vCenter Server changes the vpxuser password automatically every 30 days. Ensure that this setting meets company policy, or configure the vCenter Server password policy. See Set the vCenter Server Password Policy.

Note Make sure that password aging policy is not too short.

Check Privileges After vCenter Server Restart

Check for privilege reassignment when you restart vCenter Server. If the user or group that has the Administrator role on the root folder cannot be validated during a restart, the role is removed from that user or group. In its place, vCenter Server grants the Administrator role to the vCenter Single Sign-On administrator, administrator@vsphere.local by default. This account can then act as the vCenter Server administrator.

Reestablish a named administrator account and assign the Administrator role to that account to avoid using the anonymous vCenter Single Sign-On administrator account (administrator@vsphere.local by default).

Use High RDP Encryption Levels

On each Windows computer in the infrastructure, ensure that Remote Desktop Host Configuration settings are set to ensure the highest level of encryption appropriate for your environment.

Verify vSphere Client Certificates

Instruct users of the vSphere Client or other client applications to heed certificate verification warnings. Without certificate verification, the user might be subject of a MiTM attack.

Set the vCenter Server Password Policy

By default, vCenter Server changes the vpxuser password automatically every 30 days. You can change that value from the vSphere Client.

Procedure

1. Log in to the vCenter Server system by using the vSphere Client.
2. Select the vCenter Server system in the object hierarchy.
3. Click **Configure**.
4. Click **Advanced Settings** and click **Edit Settings**.
5. Click the **Filter** icon and enter **`VimPasswordExpirationInDays`**.
6. Set `VirtualCenter.VimPasswordExpirationInDays` to comply with your requirements.

Removing Expired or Revoked Certificates and Logs from Failed Installations

Leaving expired or revoked certificates or leaving vCenter Server installation logs for failed installation on your vCenter Server system can compromise your environment.

Removing expired or revoked certificates is required for the following reasons.

- If expired or revoked certificates are not removed from the vCenter Server system, the environment can be subject to a MiTM attack
- In certain cases, a log file that contains the database password in plain text is created on the system if vCenter Server installation fails. An attacker who breaks into the vCenter Server system, might gain access to this password and, at the same time, access to the vCenter Server database.

Limiting vCenter Server Network Connectivity

For improved security, avoid putting the vCenter Server system on any network other than a management network, and ensure that vSphere management traffic is on a restricted network. By limiting network connectivity, you limit certain types of attack.

vCenter Server requires access to a management network only. Avoid putting the vCenter Server system on other networks such as your production network or storage network, or on any network with access to the Internet. vCenter Server does not need access to the network where vMotion operates.

vCenter Server requires network connectivity to the following systems.

- All ESXi hosts.
- The vCenter Server database.
- Other vCenter Server systems (if the vCenter Server systems are part of a common vCenter Single Sign-On domain for purposes of replicating tags, permissions, and so on).
- Systems that are authorized to run management clients. For example, the vSphere Client, a Windows system where you use the PowerCLI, or any other SDK-based client.
- Infrastructure services such as DNS, Active Directory, and PTP or NTP.
- Other systems that run components that are essential to functionality of the vCenter Server system.

Use the firewall on the vCenter Server. Include IP-based access restrictions so that only necessary components can communicate with the vCenter Server system.

Evaluate the Use of Linux Clients with CLIs and SDKs

Communications between client components and a vCenter Server system or ESXi hosts are protected by SSL-based encryption by default. Linux versions of these components do not perform certificate validation. Consider restricting the use of these clients.

To improve security, you can replace the VMCA-signed certificates on the vCenter Server system and on the ESXi hosts with certificates that are signed by an enterprise or third-party CA. However, certain communications with Linux clients might still be vulnerable to machine-in-the-middle attacks. The following components are vulnerable when they run on the Linux operating system.

- ESXCLI commands
- vSphere SDK for Perl scripts
- Programs that are written using the vSphere Web Services SDK

You can relax the restriction against using Linux clients if you enforce proper controls.

- Restrict management network access to authorized systems only.
- Use firewalls to ensure that only authorized hosts are allowed to access vCenter Server.
- Use bastion hosts (jump-box systems) to ensure that the Linux clients are behind the "jump."

Examine Client Plug-Ins

vSphere Client extensions run at the same privilege level as the user who is logged in. A malicious extension can masquerade as a useful plug-in and perform harmful operations such as stealing

credentials or changing the system configuration. To increase security, use an installation that includes only authorized extensions from trusted sources.

A vCenter installation includes an extensibility framework for the vSphere Client. You can use this framework to extend the client with menu selections or toolbar icons. The extensions can provide access to vCenter add-on components or external, Web-based functionality.

Using the extensibility framework results in a risk of introducing unintended capabilities. For example, if an administrator installs a plug-in in an instance of the vSphere Client, the plug-in can run arbitrary commands with the privilege level of that administrator.

To protect against a potential compromise of your vSphere Client, examine all installed plug-ins periodically and make sure that each plug-in comes from a trusted source.

Prerequisites

You must have privileges to access the vCenter Single Sign-On service. These privileges differ from vCenter Server privileges.

Procedure

1. Log in to the vSphere Client as administrator@vsphere.local or a user with vCenter Single Sign-On privileges.
2. From the Home page, select **Administration**, then select **Client Plug-Ins** under **Solutions**.
3. Examine the list of client plug-ins.

vCenter Server Security Best Practices

Follow all best practices for securing a vCenter Server system. Additional steps help you make your vCenter Server more secure.

Configure PTP or NTP

Ensure that all systems use the same relative time source. This time source must be in sync with an agreed-upon time standard such as Coordinated Universal Time (UTC). Synchronized systems are essential for certificate validation. PTP and NTP also make it easier to track an intruder in log files. Incorrect time settings make it difficult to inspect and correlate log files to detect attacks, and make auditing inaccurate. See Synchronize the Time in vCenter Server with an NTP Server.

Restrict vCenter Server network access

Restrict access to components that are required to communicate with the vCenter Server. Blocking access from unnecessary systems reduces the potential for attacks on the operating system.

For the list of all supported ports and protocols in VMware products, including vSphere and vSAN, see the VMware Ports and Protocols Tool™ at https://ports.vmware.com/. You can search ports by VMware product, create a customized list of ports, and print or save port lists.

Configure a Bastion Host

To help protect your assets, configure a bastion host (also called a jump box) to perform elevated administrative tasks. A bastion host is a special-purpose computer that hosts a minimal number of administrative applications. All other unnecessary services are removed. The host typically resides on the management network. A bastion host increases the protection of assets through restricting login to key individuals, requiring firewall rules to log in, and adding monitoring through auditing tools.

vCenter Password Requirements and Lockout Behavior

To manage your vSphere environment, you must be aware of the vCenter Single Sign-On password policy, of vCenter Server passwords, and of lockout behavior.

This section discusses vCenter Single Sign-On passwords. See ESXi Passwords and Account Lockout for a discussion of passwords of ESXi local users.

vCenter Single Sign-On Administrator Password

The password for the administrator of vCenter Single Sign-On, administrator@vsphere.local by default, is specified by the vCenter Single Sign-On password policy. By default, this password must meet the following requirements:

- At least eight characters
- At least one lowercase character
- At least one numeric character
- At least one special character

The password for this user cannot be more than 20 characters long. Non-ASCII characters are allowed. Administrators can change the default password policy. See the *vSphere Authentication* documentation.

vCenter Server Passwords

In vCenter Server, password requirements are dictated by vCenter Single Sign-On or by the configured identity source, which can be Active Directory, OpenLDAP.

vCenter Single Sign-On Lockout Behavior

Users are locked out after a preset number of consecutive failed attempts. By default, users are locked out after five consecutive failed attempts in three minutes and a locked account is unlocked automatically after five minutes. You can change these defaults using the vCenter Single Sign-On lockout policy. See the *vSphere Authentication* documentation.

The vCenter Single Sign-On domain administrator, administrator@vsphere.local by default, is not affected by the lockout policy. The user is affected by the password policy.

Password Changes

If you know your password, you can change the password by using the `dir-cli password change` command. If you forget your password, a vCenter Single Sign-On administrator can reset your password by using the `dir-cli password reset` command.

Search the VMware Knowledge Base for information on password expiration and related topics in different versions of vSphere.

Verify Thumbprints for Legacy ESXi Hosts

In vSphere 6.0 and later, hosts are assigned VMCA certificates by default. If you change the certificate mode to thumbprint, you can continue to use thumbprint mode for legacy hosts. You can verify the thumbprints in the vSphere Client.

Note Certificates are preserved across upgrades by default.

Procedure

1. Browse to the vCenter Server in the vSphere Client inventory.
2. Click **Configure**.
3. Under **Settings**, click **General**.
4. Click **Edit**.
5. Click **SSL settings**.
6. If any of your ESXi 5.5 or earlier hosts require manual validation, compare the thumbprints listed for the hosts to the thumbprints in the host console.

   To obtain the host thumbprint, use the Direct Console User Interface (DCUI).
   a. Log in to the direct console and press F2 to access the System Customization menu.
   b. Select **View Support Information**.

      The host thumbprint appears in the column on the right.
7. If the thumbprint matches, select the **Verify** check box next to the host.

   Hosts that are not selected will be disconnected after you click **OK**.
8. Click **Save**.

Required Ports for vCenter Server

The vCenter Server system must be able to send data to every managed host and receive data from the vSphere Client. To enable migration and provisioning activities between managed

hosts, the source and destination hosts must be able to receive data from each other through predetermined TCP and UDP ports.

vCenter Server is accessed through predetermined TCP and UDP ports. If you manage network components from outside a firewall, you might be required to reconfigure the firewall to allow access on the appropriate ports. For the list of all supported ports and protocols in vSphere, see the VMware Ports and Protocols Tool™ at https://ports.vmware.com.

During installation, if a port is in use or is blocked using a denylist, the vCenter Server installer displays an error message. You must use another port number to proceed with the installation. There are internal ports that are used only for inter-process communication.

VMware uses designated ports for communication. Additionally, the managed hosts monitor designated ports for data from vCenter Server. If a built-in firewall exists between any of these elements, the installer opens the ports during the installation or upgrade process. For custom firewalls, you must manually open the required ports. If you have a firewall between two managed hosts and you want to perform source or target activities, such as migration or cloning, you must configure a means for the managed hosts to receive data.

To configure the vCenter Server system to use a different port to receive vSphere Client data, see the *vCenter Server and Host Management* documentation.

Securing Virtual Machines

5

The guest operating system that runs in the virtual machine is subject to the same security risks as a physical system. Secure virtual machines like physical machines, and follow best practices discussed in this document and in the *Security Configuration Guide* (formerly known as the *Hardening Guide*).

The *Security Configuration Guide* is available at https://core.vmware.com/security.

Read the following topics next:

- Enable or Disable UEFI Secure Boot for a Virtual Machine
- Virtual Machine Security Best Practices
- Securing Virtual Machines with Intel Software Guard Extensions
- Securing Virtual Machines with AMD Secure Encrypted Virtualization-Encrypted State

Enable or Disable UEFI Secure Boot for a Virtual Machine

UEFI Secure Boot is a security standard that helps ensure that your PC boots using only software that is trusted by the PC manufacturer. For certain virtual machine hardware versions and operating systems, you can enable secure boot just as you can for a physical machine.

In an operating system that supports UEFI secure boot, each piece of boot software is signed, including the bootloader, the operating system kernel, and operating system drivers. The virtual machine's default configuration includes several code signing certificates.

- A Microsoft certificate that is used only for booting Windows.
- A Microsoft certificate that is used for third-party code that is signed by Microsoft, such as Linux bootloaders.
- A VMware certificate that is used only for booting ESXi inside a virtual machine.

The virtual machine's default configuration includes one certificate for authenticating requests to modify the secure boot configuration, including the secure boot revocation list, from inside the virtual machine, which is a Microsoft KEK (Key Exchange Key) certificate.

VMware Tools version 10.1 or later is required for virtual machines that use UEFI secure boot. You can upgrade those virtual machines to a later version of VMware Tools when it becomes available.

For Linux virtual machines, VMware Host-Guest Filesystem is not supported in secure boot mode. Remove VMware Host-Guest Filesystem from VMware Tools before you enable secure boot.

Note If you turn on secure boot for a virtual machine, you can load only signed drivers into that virtual machine.

This task describes how to use the vSphere Client to enable and disable secure boot for a virtual machine. You can also write scripts to manage virtual machine settings. For example, you can automate changing the firmware from BIOS to EFI for virtual machines with the following PowerCLI code:

```
$vm = Get-VM TestVM

$spec = New-Object VMware.Vim.VirtualMachineConfigSpec
$spec.Firmware = [VMware.Vim.GuestOsDescriptorFirmwareType]::efi
$vm.ExtensionData.ReconfigVM($spec)
```

See *VMware PowerCLI User's Guide* for more information.

Prerequisites

You can enable secure boot only if all prerequisites are met. If prerequisites are not met, the check box is not visible in the vSphere Client.

- Verify that the virtual machine operating system and firmware support UEFI boot.
  - EFI firmware
  - Virtual hardware version 13 or later.
  - Operating system that supports UEFI secure boot.

  Note Some guest operating systems do not support changing from BIOS boot to UEFI boot without guest OS modifications. Consult your guest OS documentation before changing to UEFI boot. If you upgrade a virtual machine that already uses UEFI boot to an operating system that supports UEFI secure boot, you can enable Secure Boot for that virtual machine.
- Turn off the virtual machine. If the virtual machine is running, the check box is dimmed.

Procedure

1. Browse to the virtual machine in the vSphere Client inventory.
2. Right-click the virtual machine and select **Edit Settings**.
3. Click the **VM Options** tab, and expand **Boot Options**.
4. Under **Boot Options**, ensure that firmware is set to **EFI**.
5. Select your task.
   - Select the **Secure Boot** check box to enable secure boot.
   - Deselect the **Secure Boot** check box to disable secure boot.

6 Click **OK**.

Results

When the virtual machine boots, only components with valid signatures are allowed. The boot process stops with an error if it encounters a component with a missing or invalid signature.

Virtual Machine Security Best Practices

Following virtual machine security best practices helps ensure the integrity of your vSphere deployment.

- General Virtual Machine Protection

  A virtual machine is, in most respects, the equivalent of a physical server. Employ the same security measures in virtual machines that you do for physical systems.

- Use Templates to Deploy Virtual Machines

  When you manually install guest operating systems and applications on a virtual machine, you introduce a risk of misconfiguration. By using a template to capture a hardened base operating system image with no applications installed, you can ensure that all virtual machines are created with a known baseline level of security.

- Minimize Use of the Virtual Machine Console

  The virtual machine console provides the same function for a virtual machine that a monitor provides on a physical server. Users with access to the virtual machine console have access to virtual machine power management and removable device connectivity controls. Console access might therefore allow a malicious attack on a virtual machine.

- Prevent Virtual Machines from Taking Over Resources

  When one virtual machine consumes so much of the host resources that other virtual machines on the host cannot perform their intended functions, a Denial of Service (DoS) might occur. To prevent a virtual machine from causing a DoS, use host resource management features such as setting Shares and using resource pools.

- Disable Unnecessary Functions Inside Virtual Machines

  Any service that is running in a virtual machine provides the potential for attack. By disabling system components that are not necessary to support the application or service that is running on the system, you reduce the potential.

General Virtual Machine Protection

A virtual machine is, in most respects, the equivalent of a physical server. Employ the same security measures in virtual machines that you do for physical systems.

Follow these best practices to protect your virtual machine:

Patches and other protection

Keep all security measures up-to-date, including applying appropriate patches. It is especially important to keep track of updates for dormant virtual machines that are powered off, because it can be easy to overlook them. For example, ensure that anti-virus software, anti-spy ware, intrusion detection, and other protection are enabled for every virtual machine in your virtual infrastructure. You should also ensure that you have enough space for the virtual machine logs.

Anti-virus scans

Because each virtual machine hosts a standard operating system, you must protect it from viruses by installing anti-virus software. Depending on how you are using the virtual machine, you might also want to install a software firewall.

Stagger the schedule for virus scans, particularly in deployments with a large number of virtual machines. Performance of systems in your environment degrades significantly if you scan all virtual machines simultaneously. Because software firewalls and antivirus software can be virtualization-intensive, you can balance the need for these two security measures against virtual machine performance, especially if you are confident that your virtual machines are in a fully trusted environment.

Serial ports

Serial ports are interfaces for connecting peripherals to the virtual machine. They are often used on physical systems to provide a direct, low-level connection to the console of a server, and a virtual serial port allows for the same access to a virtual machine. Serial ports allow for low-level access, which often does not have strong controls like logging or privileges.

Use Templates to Deploy Virtual Machines

When you manually install guest operating systems and applications on a virtual machine, you introduce a risk of misconfiguration. By using a template to capture a hardened base operating system image with no applications installed, you can ensure that all virtual machines are created with a known baseline level of security.

You can use templates that can contain a hardened, patched, and properly configured operating system to create other, application-specific templates, or you can use the application template to deploy virtual machines.

Procedure

- Provide templates for virtual machine creation that contain hardened, patched, and properly configured operating system deployments.

  If possible, deploy applications in templates as well. Ensure that the applications do not depend on information specific to the virtual machine to be deployed.

What to do next

For more information about templates, see the *vSphere Virtual Machine Administration* documentation.

Minimize Use of the Virtual Machine Console

The virtual machine console provides the same function for a virtual machine that a monitor provides on a physical server. Users with access to the virtual machine console have access to virtual machine power management and removable device connectivity controls. Console access might therefore allow a malicious attack on a virtual machine.

Procedure

1 Use native remote management services, such as terminal services and SSH, to interact with virtual machines.

   Grant access to the virtual machine console only when necessary.

2 Limit the connections to the virtual machine console.

   For example, in a highly secure environment, limit the connection to one. In some environments, you can increase the limit if several concurrent connections are necessary to accomplish normal tasks.

   a In the vSphere Client, power off the virtual machine.

   b Right-click the virtual machine and select **Edit Settings**.

   c Click the **VM Options** tab, and expand **VMware Remote Console Options**.

   d Enter the maximum number of sessions, for example, **2**.

   e Click **OK**.

Prevent Virtual Machines from Taking Over Resources

When one virtual machine consumes so much of the host resources that other virtual machines on the host cannot perform their intended functions, a Denial of Service (DoS) might occur. To prevent a virtual machine from causing a DoS, use host resource management features such as setting Shares and using resource pools.

By default, all virtual machines on an ESXi host share resources equally. You can use Shares and resource pools to prevent a denial of service attack that causes one virtual machine to consume so much of the host's resources that other virtual machines on the same host cannot perform their intended functions.

Do not set limits or use resource pools until you fully understand the impact.

Procedure

1 Provision each virtual machine with just enough resources (CPU and memory) to function properly.

2 Use Shares to guarantee resources to critical virtual machines.

3 Group virtual machines with similar requirements into resource pools.

4 In each resource pool, leave Shares set to the default to ensure that each virtual machine in the pool receives approximately the same resource priority.

With this setting, a single virtual machine cannot use more than other virtual machines in the resource pool.

What to do next

See the *vSphere Resource Management* documentation for information about shares and limits.

Disable Unnecessary Functions Inside Virtual Machines

Any service that is running in a virtual machine provides the potential for attack. By disabling system components that are not necessary to support the application or service that is running on the system, you reduce the potential.

Virtual machines do not usually require as many services or functions as physical servers. When you virtualize a system, evaluate whether a particular service or function is necessary.

Note When possible, install guest operating systems using "minimal" or "core" installation modes to reduce the size, complexity, and attack surface of the guest operating system.

Procedure

- Disable unused services in the operating system.

  For example, if the system runs a file server, turn off any Web services.
- Disconnect unused physical devices, such as CD/DVD drives, floppy drives, and USB adapters.
- Disable unused functionality, such as unused display features, or VMware Shared Folders, which enables sharing of host files to the virtual machine (Host Guest File System).
- Turn off screen savers.
- Do not run the X Window system on top of Linux, BSD, or Solaris guest operating systems unless it is necessary.

Remove Unnecessary Hardware Devices

Any enabled or connected device represents a potential attack channel. Users and processes with privileges on a virtual machine can connect or disconnect hardware devices, such as network adapters and CD-ROM drives. Attackers can use this capability to breach virtual machine security. Removing unnecessary hardware devices can help prevent attacks.

An attacker with access to a virtual machine can connect a disconnected hardware device and access sensitive information on media that is left in a hardware device. The attacker can potentially disconnect a network adapter to isolate the virtual machine from its network, resulting in a denial of service.

- Do not connect unauthorized devices to the virtual machine.

- Remove unneeded or unused hardware devices.
- Disable unnecessary virtual devices from within a virtual machine.
- Ensure that only required devices are connected to a virtual machine. Virtual machines rarely use serial or parallel ports. As a rule, CD/DVD drives are connected only temporarily during software installation.

Procedure

1. Browse to the virtual machine in the vSphere Client inventory.
2. Right-click the virtual machine and click **Edit Settings**.
3. Disable hardware devices that are not required.

   Include checks for the following devices:

   - Serial ports
   - Parallel ports
   - USB controllers
   - CD-ROM drives

   Note You must use PowerCLI commands to manage floppy drive devices in vSphere 7.0 and later.

Disable Unused Display Features

Attackers can use an unused display feature as a vector for inserting malicious code into your environment. Disable features that are not in use in your environment.

Prerequisites

Power off the virtual machine.

Procedure

1. Browse to the virtual machine in the vSphere Client inventory.
2. Right-click the virtual machine and click **Edit Settings**.
3. Select **VM Options**.
4. Click **Advanced** and click **Edit Configuration**.

5 If appropriate, add or edit the following parameters.

Option	Description
svga.vgaonly	If you set this parameter to TRUE, advanced graphics functions no longer work. Do not set this parameter to TRUE with modern-day guest operating systems as they do not operate correctly. When `svga.vgaonly` is set to TRUE, only character-cell console mode is available. If you use this setting, `mks.enable3d` has no effect. **Note** Apply this setting only to virtual machines that do not need a virtualized video card.
mks.enable3d	Set this parameter to FALSE on virtual machines that do not require 3D functionality.

Disable Unexposed Features

VMware virtual machines can work both in a vSphere environment and on hosted virtualization platforms such as VMware Workstation and VMware Fusion. Certain virtual machine parameters do not need to be enabled when you run a virtual machine in a vSphere environment. Disable these parameters to reduce the potential for vulnerabilities.

Prerequisites

Turn off the virtual machine.

Procedure

1 Browse to the virtual machine in the vSphere Client inventory.

2 Right-click the virtual machine and click **Edit Settings**.

3 Select **VM Options**.

4 Click **Advanced** and click **Edit Configuration**.

5 Set the following parameters to TRUE by adding or editing them.

- `isolation.tools.memSchedFakeSampleStats.disable`
- `isolation.tools.getCreds.disable`
- `isolation.bios.bbs.disable`
- `isolation.tools.hgfsServerSet.disable`

6 Click **OK**.

Disable VMware Shared Folders Sharing Host Files to the Virtual Machine

In high-security environments, you can disable certain components to minimize the risk that an attacker can use the host guest file system (HGFS) to transfer files inside the guest operating system.

Modifying the parameters described in this section affects only the Shared Folders feature and does not affect the HGFS server running as part of tools in the guest virtual machines. Also, these parameters do not affect the auto-upgrade and VIX commands that use the tools' file transfers.

Procedure

1 Browse to the virtual machine in the vSphere Client inventory.

2 Right-click the virtual machine and click **Edit Settings**.

3 Select **VM Options**.

4 Click **Advanced** and click **Edit Configuration**.

5 Verify that the `isolation.tools.hgfsServerSet.disable` parameter is set to TRUE.

   A setting of TRUE prevents the VMX process from receiving a notification from each tool's service, daemon, or upgrader processes of its HGFS server capability.

6 (Optional) Verify that the `isolation.tools.hgfs.disable` parameter is set to TRUE.

   A setting of TRUE disables the unused VMware Shared Folders feature for sharing host files to the virtual machine.

Disable Copy and Paste Operations Between Guest Operating System and Remote Console

Copy and paste operations between the guest operating system and remote console are disabled by default. For a secure environment, retain the default setting. If you require copy and paste operations, you must enable them using the vSphere Client.

The default values for these options are set to ensure a secure environment. However, you must set them to true explicitly if you want to enable audit tools to check that the setting is correct.

Prerequisites

Turn off the virtual machine.

Procedure

1 Browse to the virtual machine in the vSphere Client inventory.

2 Right-click the virtual machine and click **Edit Settings**.

3 Select **VM Options**.

4 Click **Advanced** and click **Edit Configuration**.

5 Ensure that the following values are in the Name and Value columns, or add them.

Name	Value
isolation.tools.copy.disable	true
isolation.tools.paste.disable	true
isolation.tools.setGUIOptions.enable	false

These options override any settings made in the guest operating system's VMware Tools control panel.

6 Click **OK**.

7 (Optional) If you made changes to the configuration parameters, restart the virtual machine.

Limiting Exposure of Sensitive Data Copied to the Clipboard

Copy and paste operations are disabled by default for hosts to prevent exposing sensitive data that has been copied to the clipboard.

When copy and paste is enabled on a virtual machine running VMware Tools, you can copy and paste between the guest operating system and remote console. When the console window gains focus, processes running in the virtual machine and non-privileged users can access the virtual machine console clipboard. If a user copies sensitive information to the clipboard before using the console, the use might expose sensitive data to the virtual machine. To prevent this problem, copy and paste operations for the guest operating system are disabled by default.

It is possible to enable copy and paste operations for virtual machines if necessary.

Restrict Users from Running Commands Within a Virtual Machine

By default, a user who has the vCenter Server Administrator role can interact with files and applications within a virtual machine's guest operating system. To reduce the risk of breaching guest confidentiality, availability, or integrity, create a nonguest access role without the **Guest Operations** privilege. Assign that role to administrators who do not need virtual machine file access.

For security, be as restrictive about allowing access to the virtual data center as you are to the physical data center. Apply a custom role that disables guest access to users who require administrator privileges, but who are not authorized to interact with guest operating system files and applications.

For example, a configuration might include a virtual machine on the infrastructure that has sensitive information on it.

If tasks such as migration with vMotion require that data center administrators can access the virtual machine, disable some remote guest OS operations to ensure that those administrators cannot access sensitive information.

Prerequisites

Verify that you have **Administrator** privileges on the vCenter Server system where you create the role.

Procedure

1. Log in to the vSphere Client as a user who has **Administrator** privileges on the vCenter Server system where you want to create the role.
2. Select **Administration** and click **Roles**.
3. Click the Administrator role and click the **Clone role action** icon.
4. Enter a role name and description and click **OK**.

   For example, type **`Administrator No Guest Access`**.
5. Select the cloned role and click the **Edit role action** icon.
6. Under the **Virtual machine** privilege, deselect **Guests operations** and click **Next**.
7. Click **Finish**.

What to do next

Select the vCenter Server system or the host and assign a permission that pairs the user or group that should have the new privileges to the newly created role. Remove those users from the Administrator role.

Prevent a Virtual Machine User or Process from Disconnecting Devices

Users and processes without root or administrator privileges within virtual machines can connect or disconnect devices, such as network adapters and CD-ROM drives, and can modify device settings. To increase virtual machine security, remove these devices.

You can prevent virtual machine users in the guest OS, and processes running in the guest OS, from making any changes to the devices by changing the virtual machine advanced settings.

Prerequisites

Turn off the virtual machine.

Procedure

1. Browse to the virtual machine in the vSphere Client inventory.
2. Right-click the virtual machine and click **Edit Settings**.
3. Select **VM Options**.
4. Click **Advanced** and click **Edit Configuration**.

5 Verify that the following values are in the Name and Value columns, or add them.

Name	Value
isolation.device.connectable.disable	true
isolation.device.edit.disable	true

These settings do not affect a vSphere administrator's ability to connect or disconnect the devices attached to the virtual machine.

6 Click **OK** to close the Configuration Parameters dialog box, and click **OK** again.

Prevent Guest Operating System Processes from Sending Configuration Messages to the Host

To ensure that the guest operating system does not modify configuration settings, you can prevent these processes from writing any name-value pairs to the configuration file.

Prerequisites

Turn off the virtual machine.

Procedure

1 Browse to the virtual machine in the vSphere Client inventory.

2 Right-click the virtual machine and click **Edit Settings**.

3 Select **VM Options**.

4 Click **Advanced** and click **Edit Configuration**.

5 Click **Add Configuration Params** and enter the following values in the Name and Value columns.

Column	Value
Name	`isolation.tools.setinfo.disable`
Value	`true`

6 Click **OK** to close the Configuration Parameters dialog box, and click **OK** again.

Avoid Using Independent Nonpersistent Disks

When you use independent nonpersistent disks, successful attackers can remove any evidence that the machine was compromised by shutting down or rebooting the system. Without a persistent record of activity on a virtual machine, administrators might be unaware of an attack. Therefore, you should avoid using independent nonpersistent disks.

Procedure

- Ensure that virtual machine activity is logged remotely on a separate server, such as a syslog server or equivalent Windows-based event collector.

  If remote logging of events and activity is not configured for the guest, scsiX:Y.mode should be one of the following settings:

  - Not present
  - Not set to independent nonpersistent

Results

When nonpersistent mode is not enabled, you cannot roll a virtual machine back to a known state when you reboot the system.

Securing Virtual Machines with Intel Software Guard Extensions

vSphere enables you to configure Virtual Intel® Software Guard Extensions (vSGX) for virtual machines. Using vSGX enables you to provide additional security to your workloads.

Some modern Intel CPUs implement a security extension called Intel® Software Guard Extensions (Intel® SGX). Intel SGX is a processor-specific technology for application developers who seek to protect select code and data from disclosure or modification. Intel SGX allows user-level code to define private regions of memory, called enclaves. The enclave contents are protected such that code running outside the enclave cannot access the enclave contents.

vSGX enables virtual machines to use Intel SGX technology if available on the hardware. To use vSGX, the ESXi host must be installed on an SGX-capable CPU and SGX must be enabled in the BIOS of the ESXi host. You can use the vSphere Client to enable SGX for a virtual machine.

vSGX Overview

Virtual machines can use Intel SGX technology, if available on the hardware.

Requirements for vSGX

To use vSGX, your vSphere environment must meet these requirements:

- Virtual machine requirements:
  - EFI firmware
  - Hardware version 17 or later
- Component requirements:
  - vCenter Server 7.0 and later
  - ESXi 7.0 and later

- Guest OS support:
  - Linux
  - Windows Server 2016 (64 bit) and later
  - Windows 10 (64 bit) and later

Intel Hardware

For supported Intel hardware for vSGX, consult the VSphere Compatibility Guide at https://www.vmware.com/resources/compatibility/search.php.

You might need to turn off hyperthreading on certain CPUs to enable SGX on the ESXi host. For more information, see the VMware KB article at https://kb.vmware.com/s/article/71367.

Unsupported VMware Features on vSGX

The following features are not supported in a virtual machine when vSGX is enabled:

- vMotion/DRS migration
- Virtual machine suspend and resume
- Virtual machine snapshots (Virtual machine snapshots are supported if you do not snapshot the virtual machine's memory.)
- Fault tolerance
- Guest Integrity (GI, platform foundation for VMware AppDefense™ 1.0)

Note These VMware features are not supported due to how the Intel SGX architecture functions. They are not the result of a VMware shortcoming.

Enable vSGX on a Virtual Machine

You can enable vSGX on a virtual machine at the same time you create a virtual machine.

Prerequisites

The ESXi host must be installed on an SGX-capable CPU and SGX must be enabled in the host's BIOS. See vSGX Overview for supported Intel CPUs.

Create a virtual machine that uses hardware version 17 or later and one of the following supported guest operating systems:

- Linux
- Windows 10 (64 bit) and later
- Windows Server 2016 (64 bit) and later

Procedure

1 Connect to vCenter Server by using the vSphere Client.

2 Select an object in the inventory that is a valid parent object of a virtual machine, for example, an ESXi host or a cluster.

3 Right-click the object, select **New Virtual Machine**, and follow the prompts to create a virtual machine.

Option	Action
Select a creation type	Create a virtual machine.
Select a name and folder	Specify a name and target location.
Select a compute resource	Specify an object for which you have privileges to create virtual machines.
Select storage	In the VM storage policy, select the storage policy. Select a compatible datastore.
Select compatibility	Ensure that **ESXi 7.0 and later** is selected.
Select a guest OS	Select either Linux, Windows 10 (64-bit), or Windows Server 2016 (64-bit).
Customize hardware	Under Security devices, select the **Enable** check box for SGX. Under **VM Options > Boot Options > Firmware**, ensure that EFI is selected. Enter Enclave Page Cache (EPC) size and select Flexible Launch Control (FLC) mode accordingly.
Ready to complete	Review the information and click **Finish**.

Enable vSGX on an Existing Virtual Machine

You can enable vSGX on an existing virtual machine.

You can enable vSGX for virtual machines running on vSphere 7.0 and later.

Prerequisites

- The ESXi host must be installed on an SGX-capable CPU and SGX must be enabled in the host's BIOS. See vSGX Overview for supported Intel CPUs.
- The guest OS you use must be Linux, or Windows Server 2016 (64 bit) or later, or Windows 10 (64 bit) or later.
- The ESXi hosts running in your environment must be ESXi 7.0 or later.
- Verify that the virtual machine is turned off.
- The virtual machine must use EFI firmware.
- The virtual machine must use hardware version 17 or later.

Procedure

1 Connect to vCenter Server by using the vSphere Client.

2 Right-click the virtual machine in the inventory that you want to modify and select **Edit Settings**.

3 In the **Edit Settings** dialog box, under **Security devices**, select the **Enable** check box for SGX.

4 Enter Enclave Page Cache (EPC) size and select Flexible Launch Control (FLC) mode accordingly.

5 Under **VM Options > Boot Options > Firmware**, ensure that EFI is selected.

6 Click **OK**.

Remove vSGX from a Virtual Machine

You can remove vSGX from a virtual machine.

Procedure

1 Connect to vCenter Server by using the vSphere Client.

2 Right-click the virtual machine in the inventory that you want to modify and select **Edit Settings**.

3 In the **Edit Settings** dialog box, under **Security devices**, deselect the **Enable** check box for SGX.

4 Click **OK**.

Verify that the vSGX entry no longer appears in the virtual machine **Summary** tab in the **VM Hardware** pane.

Securing Virtual Machines with AMD Secure Encrypted Virtualization-Encrypted State

Secure Encrypted Virtualization-Encrypted State (SEV-ES) is a hardware feature enabled in recent AMD CPUs that keeps the guest operating system's memory and register state encrypted, protecting it against access from the hypervisor.

You can add SEV-ES to your virtual machines as an extra security enhancement. SEV-ES prevents CPU registers from leaking information in registers to components like the hypervisor. SEV-ES can also detect malicious modifications to a CPU register state.

AMD Secure Encrypted Virtualization-Encrypted State Overview

In vSphere 7.0 Update 1 and later, you can enable Secure Encrypted Virtualization-Encrypted State (SEV-ES) on supported AMD CPUs and guest operating systems.

Currently, SEV-ES supports only AMD EPYC 7xx2 CPUs (code named "Rome") and later CPUs, and only versions of Linux kernels that include specific support for SEV-ES.

SEV-ES Components and Architecture

The SEV-ES architecture consists of the following components.

- AMD CPU, specifically, the Platform Security Processor (PSP) that manages encryption keys and handles encryption.

- Enlightened operating system, that is, an operating system that uses guest-initiated calls to the hypervisor.
- Virtual Machine Monitor (VMM) and Virtual Machine Executable (VMX), to initialize an encrypted virtual machine state during virtual machine power-on, and also to handle calls from the guest operating system.
- VMkernel driver, to communicate unencrypted data between the hypervisor and the guest operating system.

Implementing and Managing SEV-ES on ESXi

You must first enable SEV-ES in a system's BIOS configuration. See your system's documentation for more information about accessing the BIOS configuration. After you have enabled SEV-ES in the system's BIOS, you can then add SEV-ES to a virtual machine.

You use either the vSphere Client (starting in vSphere 7.0 Update 2) or PowerCLI commands to enable and disable SEV-ES on virtual machines. You can create new virtual machines with SEV-ES, or enable SEV-ES on existing virtual machines. Privileges to manage virtual machines enabled with SEV-ES are the same as for managing regular virtual machines.

Unsupported VMware Features on SEV-ES

The following features are not supported when SEV-ES is enabled.

- System Management Mode
- vMotion
- Powered-on snapshots (however, no-memory snapshots are supported)
- Hot add or remove of CPU or memory
- Suspend/resume
- VMware Fault Tolerance
- Clones and instant clones
- Guest Integrity
- UEFI Secure Boot

Add AMD Secure Encrypted Virtualization-Encrypted State to a Virtual Machine with the vSphere Client

In vSphere 7.0 Update 2 and later, you can use the vSphere Client to add SEV-ES to a virtual machine to provide enhanced security to the guest operating system.

You can add SEV-ES to virtual machines running on ESXi 7.0 Update 1 or later.

Prerequisites

- The system must be installed with an AMD EPYC 7xx2 (code named "Rome") or later CPU and supporting BIOS.

- SEV-ES must be enabled in the BIOS.
- The number of SEV-ES virtual machines per ESXi host is controlled by the BIOS. When enabling SEV-ES in the BIOS, enter a value for the **Minimum SEV non-ES ASID** setting equal to the number of SEV-ES virtual machines plus one. For example, if you have 12 virtual machines that you want to run concurrently, enter **13**.

  Note vSphere 7.0 Update 1 supports 16 SEV-ES enabled virtual machines per ESXi host. Using a higher setting in the BIOS does not prevent SEV-ES from working, however, the limit of 16 still applies. vSphere 7.0 Update 2 supports 480 SEV-ES enabled virtual machines per ESXi host.
- The ESXi host running in your environment must be at ESXi 7.0 Update 1 or later.
- The vCenter Server must be at vSphere 7.0 Update 2 or later.
- The guest operating system must support SEV-ES.

  Currently, only Linux kernels with specific support for SEV-ES are supported.
- The virtual machine must be at hardware version 18 or later.
- The virtual machine must have the **Reserve all guest memory** option enabled, otherwise power-on fails.

Procedure

1. Connect to vCenter Server by using the vSphere Client.
2. Select an object in the inventory that is a valid parent object of a virtual machine, for example, an ESXi host or a cluster.
3. Right-click the object, select **New Virtual Machine**, and follow the prompts to create a virtual machine.

Option	Action
Select a creation type	Create a virtual machine.
Select a name and folder	Specify a name and target location.
Select a compute resource	Specify an object for which you have privileges to create virtual machines.
Select storage	In the VM storage policy, select the storage policy. Select a compatible datastore.
Select compatibility	Ensure that **ESXi 7.0 and later** is selected.
Select a guest OS	Select Linux, and select a version of Linux with specific support for SEV-ES.
Customize hardware	Under **VM Options > Boot Options > Firmware**, ensure that EFI is selected. Under **VM Options > Encryption**, select the **Enable** check box for AMD SEV-ES.
Ready to complete	Review the information and click **Finish**.

Results

The virtual machine is created with SEV-ES.

Add AMD Secure Encrypted Virtualization-Encrypted State to a Virtual Machine

You can add SEV-ES to a virtual machine to provide enhanced security to the guest operating system.

You can add SEV-ES to virtual machines running on ESXi 7.0 Update 1 or later.

Prerequisites

- The system must be installed with an AMD EPYC 7xx2 (code named "Rome") or later CPU and supporting BIOS.
- SEV-ES must be enabled in the BIOS.
- The number of SEV-ES virtual machines per ESXi host is controlled by the BIOS. When enabling SEV-ES in the BIOS, enter a value for the **Minimum SEV non-ES ASID** setting equal to the number of SEV-ES virtual machines plus one. For example, if you have 12 virtual machines that you want to run concurrently, enter **13**.

  Note vSphere 7.0 Update 1 supports 16 SEV-ES enabled virtual machines per ESXi host. Using a higher setting in the BIOS does not prevent SEV-ES from working, however, the limit of 16 still applies. vSphere 7.0 Update 2 supports 480 SEV-ES enabled virtual machines per ESXi host.
- The ESXi host running in your environment must be at ESXi 7.0 Update 1 or later.
- The guest operating system must support SEV-ES.

  Currently, only Linux kernels with specific support for SEV-ES are supported.
- The virtual machine must be at hardware version 18 or later.
- The virtual machine must have the **Reserve all guest memory** option enabled, otherwise power-on fails.
- PowerCLI 12.1.0 or later must be installed on a system with access to your environment.

Procedure

1 In a PowerCLI session, run the `Connect-VIServer` cmdlet to connect as an administrator to the vCenter Server that manages the ESXi host on which you want to add a virtual machine with SEV-ES.

```
Connect-VIServer -server vCenter_Server_ip_address -User admin_user -Password 'password'
```

2 Create the virtual machine with the `New-VM` cmdlet, specifying `-SEVEnabled $true`.

For example, first assign the host information to a variable, then create the virtual machine.

```
$vmhost = Get-VMHost -Name 10.193.25.83
New-VM -Name MyVM1 $vmhost -NumCPU 2 -MemoryMB 4 -DiskMB 4 -SEVEnabled $true
```

If you must specify the virtual hardware version, run the `New-VM` cmdlet with the `-HardwareVersion vmx-18` parameter. For example:

```
New-VM -Name MyVM1 $vmhost -NumCPU 2 -MemoryMB 4 -DiskMB 4 -SEVEnabled $true
-HardwareVersion vmx-18
```

Results

The virtual machine is created with SEV-ES.

Enable AMD Secure Encrypted Virtualization-Encrypted State on an Existing Virtual Machine with the vSphere Client

In vSphere 7.0 Update 2 and later, you can use the vSphere Client to add SEV-ES to an existing virtual machine to provide enhanced security to the guest operating system.

You can add SEV-ES to virtual machines running on ESXi 7.0 Update 1 or later.

Prerequisites

- The system must be installed with an AMD EPYC 7xx2 (code named "Rome") or later CPU and supporting BIOS.
- SEV-ES must be enabled in the BIOS.
- The number of SEV-ES virtual machines per ESXi host is controlled by the BIOS. When enabling SEV-ES in the BIOS, enter a value for the **Minimum SEV non-ES ASID** setting equal to the number of SEV-ES virtual machines plus one. For example, if you have 12 virtual machines that you want to run concurrently, enter **13**.

  Note vSphere 7.0 Update 1 supports 16 SEV-ES enabled virtual machines per ESXi host. Using a higher setting in the BIOS does not prevent SEV-ES from working, however, the limit of 16 still applies. vSphere 7.0 Update 2 supports 480 SEV-ES enabled virtual machines per ESXi host.
- The ESXi host running in your environment must be at ESXi 7.0 Update 1 or later.
- The vCenter Server must be at vSphere 7.0 Update 2 or later.
- The guest operating system must support SEV-ES.

  Currently, only Linux kernels with specific support for SEV-ES are supported.
- The virtual machine must be at hardware version 18 or later.

- The virtual machine must have the **Reserve all guest memory** option enabled, otherwise power-on fails.
- Ensure that the virtual machine is powered off.

Procedure

1. Connect to vCenter Server by using the vSphere Client.
2. Right-click the virtual machine in the inventory that you want to modify and select **Edit Settings**.
3. Under **VM Options > Boot Options > Firmware**, ensure that EFI is selected.
4. In the **Edit Settings** dialog box, under **VM Options > Encryption**, select the **Enable** check box for AMD SEV-ES.
5. Click **OK**.

Results

SEV-ES is added to the virtual machine.

Enable AMD Secure Encrypted Virtualization-Encrypted State on an Existing Virtual Machine

You can add SEV-ES to an existing virtual machine to provide enhanced security to the guest operating system.

You can add SEV-ES to virtual machines running on ESXi 7.0 Update 1 or later.

Prerequisites

- The system must be installed with an AMD EPYC 7xx2 (code named "Rome") or later CPU and supporting BIOS.
- SEV-ES must be enabled in the BIOS.
- The number of SEV-ES virtual machines per ESXi host is controlled by the BIOS. When enabling SEV-ES in the BIOS, enter a value for the **Minimum SEV non-ES ASID** setting equal to the number of SEV-ES virtual machines plus one. For example, if you have 12 virtual machines that you want to run concurrently, enter **13**.

  Note vSphere 7.0 Update 1 supports 16 SEV-ES enabled virtual machines per ESXi host. Using a higher setting in the BIOS does not prevent SEV-ES from working, however, the limit of 16 still applies. vSphere 7.0 Update 2 supports 480 SEV-ES enabled virtual machines per ESXi host.

- The ESXi host running in your environment must be ESXi 7.0 Update 1 or later.
- The guest operating system must support SEV-ES.

  Currently, only Linux kernels with specific support for SEV-ES are supported.

- The virtual machine must be at hardware version 18 or later.
- The virtual machine must have the **Reserve all guest memory** option enabled, otherwise power-on fails.
- PowerCLI 12.1.0 or later must be installed on a system with access to your environment.
- Ensure that the virtual machine is powered off.

Procedure

1. In a PowerCLI session, run the `Connect-VIServer` cmdlet to connect as an administrator to the vCenter Server that manages the ESXi host with the virtual machine to which you want to add SEV-ES.

   For example:

   ```
   Connect-VIServer -server vCenter_Server_ip_address -User admin_user -Password 'password'
   ```

2. Add SEV-ES to the virtual machine with the `Set-VM` cmdlet, specifying `-SEVEnabled $true`.

   For example:

   ```
   $vmhost = Get-VMHost -Name 10.193.25.83
   Set-VM -Name MyVM2 $vmhost -SEVEnabled $true
   ```

   If you must specify the virtual hardware version, run the `Set-VM` cmdlet with the `-HardwareVersion vmx-18` parameter. For example:

   ```
   Set-VM -Name MyVM2 $vmhost -SEVEnabled $true -HardwareVersion vmx-18
   ```

Results

SEV-ES is added to the virtual machine.

Disable AMD Secure Encrypted Virtualization-Encrypted State on a Virtual Machine with the vSphere Client

In vSphere 7.0 Update 2 and later, you can use the vSphere Client to disable SEV-ES on a virtual machine.

Prerequisites

- Ensure that the virtual machine is powered off.

Procedure

1. Connect to vCenter Server by using the vSphere Client.
2. Right-click the virtual machine in the inventory that you want to modify and select **Edit Settings**.
3. In the **Edit Settings** dialog box, under **VM Options > Encryption**, deselect the **Enable** check box for AMD SEV-ES.

4 Click **OK**.

Results

SEV-ES is disabled on the virtual machine.

Disable AMD Secure Encrypted Virtualization-Encrypted State on a Virtual Machine

You can disable SEV-ES on a virtual machine.

Prerequisites

- Ensure that the virtual machine is powered off.
- PowerCLI 12.1.0 or later must be installed on a system that has access to your environment.

Procedure

1 In a PowerCLI session, run the `Connect-VIServer` cmdlet to connect as an administrator to the vCenter Server that manages the ESXi host with the virtual machine from which you want to remove SEV-ES.

For example:

```
Connect-VIServer -server vCenter_Server_ip_address -User admin_user -Password 'password'
```

2 Disable SEV-ES on the virtual machine with the `Set-VM` cmdlet, specifying `-SEVEnabled $false`.

For example, first assign the host information to a variable, then disable SEV-ES for the virtual machine.

```
$vmhost = Get-VMHost -Name 10.193.25.83
Set-VM -Name MyVM2 $vmhost -SEVEnabled $false
```

Results

SEV-ES is disabled on the virtual machine.

Virtual Machine Encryption 6

With vSphere Virtual Machine Encryption, you can encrypt your sensitive workloads in an even more secure way. Access to encryption keys can be made conditional to the ESXi host being in a trusted state.

Before you can start with virtual machine encryption tasks, you must set up a key provider. The following key provider types are available.

Table 6-1. vSphere Key Providers

Key Provider	Description	For More Information
Standard key provider	Available in vSphere 6.5 and later, the standard key provider uses vCenter Server to request keys from an external key server. The key server generates and stores the keys, and passes them to vCenter Server for distribution.	See Chapter 7 Configuring and Managing a Standard Key Provider.
Trusted key provider	Available in vSphere 7.0 and later, the vSphere Trust Authority trusted key provider makes access to the encryption keys conditional to the attestation state of a workload cluster. vSphere Trust Authority requires an external key server.	See Chapter 9 vSphere Trust Authority.
VMware vSphere® Native Key Provider™	Available in vSphere 7.0 Update 2 and later, vSphere Native Key Provider is included in all vSphere editions and does not require an external key server.	See Chapter 8 Configuring and Managing vSphere Native Key Provider.

Read the following topics next:

- Comparison of vSphere Key Providers
- How vSphere Virtual Machine Encryption Protects Your Environment
- vSphere Virtual Machine Encryption Components
- Encryption Process Flow
- Virtual Disk Encryption

- Virtual Machine Encryption Errors
- Prerequisites and Required Privileges for Encryption Tasks
- Encrypted vSphere vMotion
- Encryption Best Practices, Caveats, and Interoperability
- Key Persistence Overview

Comparison of vSphere Key Providers

A high-level overview of the capabilities of the vSphere key providers requires your attention to help plan your encryption strategy.

In general, there is little difference in feature or product support between key provider daily operation. Though key providers look and behave similarly, you might have requirements and regulations to consider when choosing a key provider, as shown in the following table.

Table 6-2. Key Provider Considerations

Key Provider	External Key Server Required?	Quick Setup?	Works Only with vSphere?
Standard key provider	Yes	No	No
Trusted key provider	Yes	No	No
vSphere Native Key Provider	No	Yes	Yes

Encryption Features

The following encryption features are supported by each key provider type.

- Rekey using the same key provider or to another key provider
- Rotate keys
- Virtual Trusted Platform Module (vTPM)
- Disk encryption
- vSphere Virtual Machine Encryption
- Co-existence with other key providers
- Upgrade to a different key provider

vSphere Features

The following describes key provider support for some important vSphere features.

- Encrypted vSphere vMotion: Supported by all key provider types. The same key provider must be available on the destination host. See Encrypted vSphere vMotion.

- vCenter Server File-based Backup and Restore: Standard key provider and vSphere Native Key Provider support vCenter Server file-based backup and restore. Because most vSphere Trust Authority configuration information is stored on the ESXi hosts, the vCenter Server file-based backup mechanism does not back up this information. To ensure the configuration information for your vSphere Trust Authority deployment is saved, see Backing Up the vSphere Trust Authority Configuration.

VMware Products

The following table compares key provider support for some VMware products.

Table 6-3. Comparison of Support for VMware Products

Key Provider	vSAN	Site Recovery Manager	vSphere Replication
Standard key provider	Yes	Yes	Yes
Trusted key provider	Yes	Yes If the same vSphere Trust Authority services configuration is available on the recovery side, then SRM with array-based replication is supported.	No
vSphere Native Key Provider	Yes	Yes	Yes

Required Hardware

The following table compares some minimum key provider hardware requirements.

Table 6-4. Comparison of Required Hardware

Key Provider	TPM on ESXi Host
Standard key provider	Not required
Trusted key provider	Required on Trusted Hosts (hosts in the Trusted Cluster). **Note** Currently, the ESXi hosts in the Trust Authority Cluster do not require a TPM. However, as a matter of best practice, consider installing new ESXi hosts with TPMs.
vSphere Native Key Provider	Not required vSphere Native Key Provider availability can optionally be restricted to hosts with a TPM.

Key Provider Naming

vSphere uses a key provider name to look up a key identifier. If two key providers have the same name, vSphere assumes that they are equivalent and have access to the same keys. Each logical key provider, regardless of its type (Standard, Trusted, and Native Key Provider), must have a unique name across all vCenter Server systems.

In a few instances, you configure the same key provider across multiple vCenter Server systems, such as:

- Migrating encrypted virtual machines between vCenter Server systems
- Setting up a vCenter Server as a disaster recovery site

How vSphere Virtual Machine Encryption Protects Your Environment

Regardless of which key provider you use, with vSphere Virtual Machine Encryption you can create encrypted virtual machines and encrypt existing virtual machines. Because all virtual machine files with sensitive information are encrypted, the virtual machine is protected. Only administrators with encryption privileges can perform encryption and decryption tasks.

Important ESXi Shell users also have cryptographic operation privileges. For more information, see Prerequisites and Required Privileges for Encryption Tasks.

What Storage Does vSphere Virtual Machine Encryption Support

vSphere Virtual Machine Encryption works with any supported storage type (NFS, iSCSI, Fibre Channel, direct-attached storage, and so on), including VMware vSAN. For more information about using encryption on a vSAN cluster, see *Administering VMware vSAN* documentation.

vSphere Virtual Machine Encryption and vSAN use the same encryption libraries but they have different profiles. VM Encryption is a per-VM encryption and vSAN is a datastore level encryption.

vSphere Encryption Keys and Key Providers

vSphere uses two levels of encryption in the form of a Key Encryption Key (KEK) and a Data Encryption Key (DEK). Briefly, an ESXi host generates a DEK to encrypt virtual machines and disks. The KEK is provided by a key server, and encrypts (or "wraps") the DEK. The KEK is encrypted using the AES256 algorithm and the DEK is encrypted using the XTS-AES-256 algorithm. Depending on the type of key provider, different methods are used to create and manage the DEK and KEK.

Standard key provider operates as follows.

1. The ESXi host generates and uses internal keys to encrypt virtual machines and disks. These keys are used as DEKs.

2 vCenter Server requests keys from the key server (KMS). These keys are used as the KEKs. vCenter Server stores only the ID of each KEK, but not the key itself.

3 ESXi uses the KEK to encrypt the internal keys, and stores the encrypted internal key on disk. ESXi does not store the KEK on disk. If a host reboots, vCenter Server requests the KEK with the corresponding ID from the key server and makes it available to ESXi. ESXi can then decrypt the internal keys as needed.

vSphere Trust Authority trusted key provider operates as follows.

1 The vCenter Server of the Trusted Cluster checks if the default trusted key provider is accessible to the ESXi host where the encrypted virtual machine is to be created.

2 The vCenter Server of the Trusted Cluster adds the trusted key provider to the virtual machine ConfigSpec.

3 The virtual machine creation request is sent to the ESXi host.

4 If an attestation token is not already available to the ESXi host, it requests one from the Attestation Service.

5 The Key Provider Service validates the attestation token and creates a KEK to be sent to the ESXi host. The KEK is wrapped (encrypted) with the primary key that is configured on the key provider. Both the KEK ciphertext and KEK plaintext are returned to the Trusted Host.

6 The ESXi host generates a DEK to encrypt the virtual machine disks.

7 The KEK is used to wrap the DEK generated by the ESXi host, and the ciphertext from the key provider is stored alongside the encrypted data.

8 The virtual machine is encrypted and written to storage.

Note If you delete or unregister an encrypted virtual machine, the ESXi host and the cluster remove the KEK from cache. The ESXi host can no longer use the KEK. This behavior is the same for standard key providers and trusted key providers.

vSphere Native Key Provider operates as follows.

1 When you create the key provider, vCenter Server generates a primary key and pushes it to the ESXi hosts in the cluster. (No external key server is involved.)

2 The ESXi hosts generate a DEK on demand.

3 When you perform an encryption activity, data is encrypted with the DEK.

   Encrypted DEKs are stored alongside the encrypted data.

4 When you decrypt data, the primary key is used to decrypt the DEK, and then the data.

What Is Encrypted

vSphere Virtual Machine Encryption supports encryption of virtual machine files, virtual disk files, and core dump files.

Virtual machine files

Most virtual machine files, in particular, guest data that are not stored in the VMDK file, are encrypted. This set of files includes but is not limited to the NVRAM, VSWP, and VMSN files. The key from the key provider unlocks an encrypted bundle in the VMX file that contains internal keys and other secrets. The key retrieval works as follows, depending on the key provider:

- Standard key provider: vCenter Server manages the keys from the key server and ESXi hosts cannot directly access the key provider. The hosts wait for vCenter Server to push the keys.
- Trusted key provider and vSphere Native Key Provider: The ESXi hosts directly access the key providers, and so fetch the requested keys either from the vSphere Trust Authority service directly or the vSphere Native Key Provider.

When you use the vSphere Client to create an encrypted virtual machine, you can encrypt and decrypt virtual disks separate from virtual machine files. All virtual disks are encrypted by default. For other encryption tasks, such as encrypting an existing virtual machine, you can encrypt and decrypt virtual disks separate from virtual machine files.

Note You cannot associate an encrypted virtual disk with a virtual machine that is not encrypted.

Virtual disk files

Data in an encrypted virtual disk (VMDK) file is never written in cleartext to storage or physical disk, and is never transmitted over the network in cleartext. The VMDK descriptor file is mostly cleartext, but contains a key ID for the KEK and the internal key (DEK) in the encrypted bundle.

You can use the vSphere API to perform either a shallow recrypt operation with a new KEK or deep recrypt operation with a new internal key.

Core dumps

Core dumps on an ESXi host that has encryption mode enabled are always encrypted. See vSphere Virtual Machine Encryption and Core Dumps. Core dumps on the vCenter Server system are not encrypted. Protect access to the vCenter Server system.

Note For information on some limitations concerning devices and features that vSphere Virtual Machine Encryption can interoperate with, see Virtual Machine Encryption Interoperability.

What Is Not Encrypted

Some of the files that are associated with a virtual machine are not encrypted or partially encrypted.

Log files

Log files are not encrypted because they do not contain sensitive data.

Virtual machine configuration files

Most of the virtual machine configuration information, stored in the VMX and VMSD files, is not encrypted.

Virtual disk descriptor file

To support disk management without a key, most of the virtual disk descriptor file is not encrypted.

Who Can Perform Cryptographic Operations

Only users that are assigned the **Cryptographic Operations** privileges can perform cryptographic operations. The privilege set is fine grained. The default Administrator system role includes all **Cryptographic Operations** privileges. The No Cryptography Administrator role supports all Administrator privileges except for the **Cryptographic Operations** privileges.

In addition to using the **Cryptographer.*** privileges, vSphere Native Key Provider can use the **Cryptographer.ReadKeyServersInfo** privilege, which is specific to vSphere Native Key Providers.

See Cryptographic Operations Privileges for more information.

You can create additional custom roles, for example, to allow a group of users to encrypt virtual machines but to prevent them from decrypting virtual machines.

How Can I Perform Cryptographic Operations

The vSphere Client supports many of the cryptographic operations. For other tasks, you can use the vSphere API.

Table 6-5. Interfaces for Performing Cryptographic Operations

Interface	Operations	Information
vSphere Client	Create encrypted virtual machine Encrypt and decrypt virtual machines	This book
PowerCLI	Create encrypted virtual machine Encrypt and decrypt virtual machines Configure vSphere Trust Authority	*VMware PowerCLI Cmdlets Reference*

Table 6-5. Interfaces for Performing Cryptographic Operations (continued)

Interface	Operations	Information
vSphere Web Services SDK	Create encrypted virtual machine Encrypt and decrypt virtual machines Perform a deep recrypt of a virtual machine (use a different DEK) Perform a shallow recrypt of a virtual machine (use a different KEK)	*vSphere Web Services SDK Programming Guide* *vSphere Web Services API Reference*
`crypto-util`	Decrypt encrypted core dumps Check whether files are encrypted Perform other management tasks directly on the ESXi host	Command-line help vSphere Virtual Machine Encryption and Core Dumps

Recrypting Virtual Machines

You can recrypt a virtual machine with new keys, for example, in case a key expires or becomes compromised. The following options are available.

- A deep recrypt, which replaces both the Disk Encryption Key (DEK) and Key Encryption Key (KEK)
- A shallow recrypt, which replaces only the KEK

You must perform a recrypt of a virtual machine by using the API. See *vSphere Web Services SDK Programming Guide*.

A deep recrypt requires that the virtual machine is powered off and contains no snapshots. You can perform a shallow recrypt operation while the virtual machine is powered on, and if the virtual machine has snapshots present. Shallow recrypt of an encrypted virtual machine with snapshots is permitted only on a single snapshot branch (disk chain). Multiple snapshot branches are not supported. Also, shallow recrypt is not supported on a linked clone of a virtual machine or disk. If the shallow recrypt fails before updating all links in the chain with the new KEK, you can still access the encrypted virtual machine if you have the old and new KEKs. However, it is best to reissue the shallow recrypt operation before performing any snapshot operations.

vSphere Virtual Machine Encryption Components

Depending on which key provider you use, an external key server, the vCenter Server system, and your ESXi hosts are potentially contributing to the encryption solution.

The following components comprise vSphere Virtual Machine Encryption:

- An external key server, also called a KMS (not required for vSphere Native Key Provider)
- vCenter Server
- ESXi hosts

Key Server

The key server is a Key Management Interoperability Protocol (KMIP) management server that is associated with a key provider. A standard key provider and a trusted key provider require a key server. vSphere Native Key Provider does not require a key server. The following table describes the differences in key provider and key server interaction.

Table 6-6. Key Providers and Key Server Interaction

Key Provider	Interaction with Key Server
Standard key provider	A standard key provider uses vCenter Server to request keys from a key server. The key server generates and stores the keys, and passes them to vCenter Server for distribution to the ESXi hosts.
Trusted key provider	A trusted key provider uses a Key Provider Service that enables the trusted ESXi hosts to fetch the keys directly. See What Is the vSphere Trust Authority Key Provider Service.
vSphere Native Key Provider	vSphere Native Key Provider does not require a key server. vCenter Server generates a primary key and pushes it to the ESXi hosts. The ESXi hosts then generate data encryption keys (even when not connected to vCenter Server). See vSphere Native Key Provider Overview.

You can use the vSphere Client or the vSphere API to add key provider instances to the vCenter Server system. If you use multiple key provider instances, all instances must be from the same vendor and must replicate keys.

If your environment uses different key server vendors in different environments, you can add a key provider for each key server and specify a default key provider. The first key provider that you add becomes the default key provider. You can explicitly specify the default later.

As a KMIP client, vCenter Server uses the Key Management Interoperability Protocol (KMIP) to make it easy to use the key server of your choice.

vCenter Server

The following table describes the role of vCenter Server in the encryption process.

Table 6-7. Key Providers and vCenter Server

Key Provider	Role of vCenter Server	How Are Privileges Checked
Standard key provider	Only vCenter Server has the credentials for logging in to the key server. Your ESXi hosts do not have those credentials. vCenter Server obtains keys from the key server and pushes them to the ESXi hosts. vCenter Server does not store the key server keys, but keeps a list of key IDs.	vCenter Server checks the privileges of users who perform cryptographic operations.
Trusted key provider	vSphere Trust Authority removes the need for vCenter Server to request keys from the key server, and makes access to the encryption keys conditional to the attestation state of a workload cluster. You must use separate vCenter Server systems for the Trusted Cluster and Trust Authority Cluster.	vCenter Server checks the privileges of users who perform cryptographic operations. Only users who are members of the TrustedAdmins SSO group can perform administrative operations.
vSphere Native Key Provider	The vCenter Server generates the keys.	vCenter Server checks the privileges of users who perform cryptographic operations.

You can use the vSphere Client to assign cryptographic operation privileges or to assign the **No cryptography administrator** custom role to groups of users. See Prerequisites and Required Privileges for Encryption Tasks.

vCenter Server adds cryptography events to the list of events that you can view and export from the vSphere Client Event Console. Each event includes the user, time, key ID, and cryptographic operation.

The keys that come from the key server are used as key encryption keys (KEKs).

ESXi Hosts

ESXi hosts are responsible for several aspects of the encryption workflow.

Table 6-8. ESXi Hosts

Key Provider	ESXi Host Aspects
Standard key provider	■ vCenter Server pushes keys to an ESXi host when the host needs a key. The host must have encryption mode enabled. The current user's role must include cryptographic operation privileges. See Prerequisites and Required Privileges for Encryption Tasks and Cryptographic Operations Privileges. ■ Ensuring that guest data for encrypted virtual machines is encrypted when stored on disk. ■ Ensuring that guest data for encrypted virtual machines is not sent over the network without encryption.
Trusted key provider	The ESXi hosts run vSphere Trust Authority services, depending on if they are Trusted Hosts or Trust Authority Hosts. Trusted ESXi hosts run workload virtual machines that can be encrypted using key providers published by the Trust Authority Hosts. See Trusted Infrastructure Overview.
vSphere Native Key Provider	The ESXi hosts fetch keys directly from the vSphere Native Key Provider.

The keys that the ESXi host generates are called internal keys in this document. These keys typically act as data encryption keys (DEKs).

Encryption Process Flow

After you set up a key provider, users with the required privileges can create encrypted virtual machines and disks. Those users can also encrypt existing virtual machines and decrypt encrypted virtual machines, and add Virtual Trusted Platform Modules (vTPMs) to virtual machines.

Depending on the key provider type, the process flow can involve a key server, the vCenter Server, and the ESXi host.

Standard Key Provider Encryption Process Flow

During the encryption process, different vSphere components interact as follows.

1 When the user performs an encryption task, for example, creating an encrypted virtual machine, vCenter Server requests a new key from the default key server. This key is used as the KEK.

2 vCenter Server stores the key ID and passes the key to the ESXi host. If the ESXi host is part of a cluster, vCenter Server sends the KEK to each host in the cluster.

   The key itself is not stored on the vCenter Server system. Only the key ID is known.

3 The ESXi host generates internal keys (DEKs) for the virtual machine and its disks. It keeps the internal keys in memory only, and uses the KEKs to encrypt internal keys.

Unencrypted internal keys are never stored on disk. Only encrypted data is stored. Because the KEKs come from the key server, the host continues to use the same KEKs.

4 The ESXi host encrypts the virtual machine with the encrypted internal key.

Any hosts that have the KEK and that can access the encrypted key file can perform operations on the encrypted virtual machine or disk.

Trusted Key Provider Encryption Process Flow

The vSphere Trust Authority encryption process flow includes the vSphere Trust Authority services, the trusted key providers, the vCenter Server, and the ESXi hosts.

Encrypting a virtual machine with a trusted key provider looks the same as the virtual machine encryption user experience when using a standard key provider. Virtual machine encryption under vSphere Trust Authority continues to rely on either virtual machine encryption storage policies, or the presence of a vTPM device, to decide when to encrypt a virtual machine. You still use a default configured key provider (called a KMS cluster in vSphere 6.5 and 6.7) when encrypting a virtual machine from the vSphere Client. And, you can still use the APIs in a similar way to specify the key provider manually. The existing Cryptographic privileges added for vSphere 6.5 are still relevant in vSphere 7.0 for vSphere Trust Authority.

The encryption process for the trusted key provider has some important differences from the standard key provider:

- Trust Authority administrators do not specify information directly when setting up a key server for a vCenter Server instance, and they do not establish the key server trust. Instead, vSphere Trust Authority publishes trusted key providers that the Trusted Hosts can use.
- vCenter Server no longer pushes keys to ESXi hosts and instead it can treat each trusted key provider as a single top-level key.
- Only Trusted Hosts can request encryption operations from Trust Authority Hosts.

vSphere Native Key Provider Encryption Process Flow

vSphere Native Key Provider is included in vSphere starting with the 7.0 Update 2 release. When you configure a vSphere Native Key Provider, vCenter Server pushes a primary key to all ESXi hosts in the cluster. Likewise, if you update or delete a vSphere Native Key Provider, the change is pushed to the hosts in the cluster. The encryption process flow is similar to how a trusted key provider works. The difference is that vSphere Native Key Provider generates the keys and wraps them with the primary key, then hands them back to perform encryption.

Custom Attributes for Key Servers

The Key Management Interoperability Protocol (KMIP) supports adding custom attributes intended for vendor-specific purposes. Custom attributes enable you to more specifically identify keys stored in your key server. vCenter Server adds the following custom attributes for virtual machine keys and host keys.

Table 6-9. Virtual Machine Encryption Custom Attributes

Custom Attribute	Value
x-Vendor	VMware, Inc.
x-Product	VMware vSphere
x-Product_Version	vCenter Server version
x-Component	Virtual Machine
x-Name	Virtual machine name (gathered from ConfigInfo or ConfigSpec)
x-Identifier	Virtual machine's instanceUuid (gathered from ConfigInfo or ConfigSpec)

Table 6-10. Host Encryption Custom Attributes

Custom Attribute	Value
x-Vendor	VMware, Inc.
x-Product	VMware vSphere
x-Product_Version	vCenter Server version
x-Component	ESXi Server
x-Name	Host name
x-Identifier	Host's hardware Uuid

vCenter Server adds the `x-Vendor`, `x-Product`, and `x-Product_Version` attributes when the key server creates a key. When the key is used to encrypt a virtual machine or host, vCenter Server sets the `x-Component`, `x-Identifier`, and `x-Name` attributes. You might be able to view these custom attributes in your key server user interface. Check with your key server vendor.

Both the host key and virtual machine key have the six custom attributes. `x-Vendor`, `x-Product`, and `x-Product_Version` might be the same for both keys. These attributes are set when the key is generated. Depending on if the key is for a virtual machine or a host, it might have `x-Component`, `x-Identifier`, and `x-Name` attributes appended.

Key Errors

When an error occurs sending keys from the key server to an ESXi host, vCenter Server generates a message in the event log for the following events:

- Adding keys to the ESXi host failed due to host connection or host support issues.
- Getting keys from the key server failed due to key missing in the key server.
- Getting keys from the key server failed due to the key server connection.

Decrypting Encrypted Virtual Machines

If you later want to decrypt an encrypted virtual machine, you change its storage policy. You can change the storage policy for the virtual machine and all disks. If you want to decrypt individual components, decrypt selected disks first, then decrypt the virtual machine by changing the storage policy for VM Home. Both keys are required for decryption of each component. See Decrypt an Encrypted Virtual Machine or Virtual Disk.

Virtual Disk Encryption

When you create an encrypted virtual machine from the vSphere Client, you can decide which disks to exclude from encryption. You can later add disks and set their encryption policies. You cannot add an encrypted disk to a virtual machine that is not encrypted, and you cannot encrypt a disk if the virtual machine is not encrypted.

Encryption for a virtual machine and its disks is controlled through storage policies. The storage policy for VM Home governs the virtual machine itself, and each virtual disk has an associated storage policy.

- Setting the storage policy of VM Home to an encryption policy encrypts only the virtual machine itself.
- Setting the storage policy of VM Home and all the disks to an encryption policy encrypts all components.

Consider the following use cases.

Table 6-11. Virtual Disk Encryption Use Cases

Use Case	Details
Create an encrypted virtual machine.	If you add disks while creating an encrypted virtual machine, the disks are encrypted by default. You can change the policy not to encrypt one or more of the disks. After virtual machine creation, you can explicitly change the storage policy for each disk. See Change the Encryption Policy for Virtual Disks.
Encrypt a virtual machine.	To encrypt an existing virtual machine, you change its storage policy. You can change the storage policy for the virtual machine and all virtual disks. To encrypt just the virtual machine, you can specify an encryption policy for VM Home and select a different storage policy, such as Datastore Default, for each virtual disk. See Create an Encrypted Virtual Machine.
Add an existing unencrypted disk to an encrypted virtual machine (encryption storage policy).	Fails with an error. You have to add the disk with the default storage policy, but can later change the storage policy. See Change the Encryption Policy for Virtual Disks.
Add an existing unencrypted disk to an encrypted virtual machine with a storage policy that does not include encryption, for example Datastore Default.	The disk uses the default storage policy. You can explicitly change the storage policy after adding the disk if you want an encrypted disk. See Change the Encryption Policy for Virtual Disks.

Table 6-11. Virtual Disk Encryption Use Cases (continued)

Use Case	Details
Add an encrypted disk to an encrypted virtual machine. VM Home storage policy is Encryption.	When you add the disk, it remains encrypted. The vSphere Client displays the size and other attributes, including encryption status.
Add an existing encrypted disk to an unencrypted virtual machine.	Prior to vSphere 7.0 Update 3i, you cannot encrypt the virtual disk of an unencrypted virtual machine. In vSphere Update 3i and later, the vSphere Client prompts if you want to reconfigure the unencrypted virtual machine with the encrypted disk by encrypting VM Home.
Register an encrypted virtual machine.	If you remove an encrypted virtual machine from vCenter Server but do not delete it from disk, you can return it to the vCenter Server inventory by registering the VM's virtual machine configuration (.vmx) file. To register the encrypted VM, the user must have the **Cryptographic operations.Register VM** privilege. If the VM was encrypted using a standard key provider, when the encrypted VM is registered, vCenter Server pushes the required keys to the ESXi host. If the user registering the VM does not have the **Cryptographic operations.Register VM** privilege, vCenter Server locks the VM upon registration, and the VM is not usable until it is unlocked. If the VM was encrypted using a trusted key provider or vSphere Native Key Provider, when the encrypted VM is registered, vCenter Server no longer pushes keys to the ESXi host. Instead, the keys are fetched from the host when the VM is registered. If the user registering the VM does not have the **Cryptographic operations.Register VM** privilege, vCenter Server does not permit the operation.

Virtual Machine Encryption Errors

If vCenter Server detects a critical error with virtual machine encryption, it creates an event. You can view these events to help troubleshoot and resolve encryption errors.

vCenter Server creates events for the following virtual machine encryption critical errors.

- Failure to generate a KEK.
- Insufficient disk space on the datastore to create an encrypted virtual machine.
- Insufficient user privilege to initiate encryption operation.
- The specified key is missing on the key provider and so the ESXi host key is renewed with a new key.
- An error occurred on the key provider with the specified key and so the ESXi host key is renewed with a new key.

Prerequisites and Required Privileges for Encryption Tasks

Encryption tasks are possibly only in environments that include vCenter Server. In addition, the ESXi host must have encryption mode enabled for most encryption tasks. The user who performs the task must have the appropriate privileges. A set of **Cryptographic Operations** privileges allows fine-grained control. If virtual machine encryption tasks require a change to the host encryption mode, additional privileges are required.

Note vSphere Trust Authority has additional prerequisites and required privileges. See Prerequisites and Required Privileges for vSphere Trust Authority.

Cryptography Privileges and Roles

By default, the user with the vCenter Server Administrator role has all privileges. The **No cryptography administrator** role does not have the following privileges that are required for cryptographic operations.

Important ESXi Shell users also have cryptographic operation privileges.

- Add **Cryptographic Operations** privileges.
- **Global.Diagnostics**
- **Host.Inventory.Add host to cluster**
- **Host.Inventory.Add standalone host**
- **Host.Local operations.Manage user groups**

You can assign the **No cryptography administrator** role to vCenter Server administrators that do not need **Cryptographic Operations** privileges.

To impose more limits on what users can do, you can clone the **No cryptography administrator** role and create a custom role with only some of the **Cryptographic Operations** privileges. For example, you can create a role that allows users to encrypt but not to decrypt virtual machines. See Using Roles to Assign Privileges.

Host Encryption Mode

Host encryption mode determines if an ESXi host is ready to accept cryptographic material for encrypting virtual machines and virtual disks. Before any cryptographic operations can occur on a host, host encryption mode must be enabled. Host encryption mode is often enabled automatically when it is required, but you can enable it explicitly. You can check and explicitly set the current host encryption mode from the vSphere Client or by using the vSphere API.

When host encryption mode is enabled, vCenter Server installs a host key on the host, which ensures that the host is cryptographically "safe." With the host key in place, other cryptographic operations can proceed, including vCenter Server obtaining keys from the key provider and pushing them to the ESXi hosts.

In "safe" mode, user worlds (that is, hostd) and encrypted virtual machines have their core dumps encrypted. Unencrypted virtual machines do not have their core dumps encrypted.

For more information about encrypted core dumps and how they are used by VMware Technical Support, see the VMware knowledge base article at http://kb.vmware.com/kb/2147388.

For instructions, see Enable Host Encryption Mode Explicitly.

After Host encryption mode is enabled, it cannot be disabled easily. See Disable Host Encryption Mode Using the API.

Automatic changes occur when encryption operations attempt to enable host encryption mode. For example, assume that you add an encrypted virtual machine to a standalone host. Host encryption mode is not enabled. If you have the required privileges on the host, encryption mode changes to enabled automatically.

Assume that a cluster has three ESXi hosts, host A, B, and C. You create an encrypted virtual machine on host A. What happens depends on several factors.

- If hosts A, B, and C already have encryption enabled, you need only **Cryptographic operations.Encrypt new** privileges to create the virtual machine.
- If hosts A and B are enabled for encryption and C is not enabled, the system proceeds as follows.
  - Assume that you have both the **Cryptographic operations.Encrypt new** and the **Cryptographic operations.Register host** privileges on each host. In that case, the virtual machine creation process enables encryption on host C. The encryption process enables host encryption mode on host C, and pushes the key to each host in the cluster.

    For this case, you can also explicitly enable host encryption on host C.
  - Assume that you have only **Cryptographic operations.Encrypt new** privileges on the virtual machine or virtual machine folder. In that case, virtual machine creation succeeds and the key becomes available on host A and host B. Host C remains disabled for encryption and does not have the virtual machine key.
- If none of the hosts has encryption enabled, and you have **Cryptographic operations.Register host** privileges on host A, then the virtual machine creation process enables host encryption on that host. Otherwise, an error results.
- You can also use the vSphere API to set the encryption mode of a cluster to "force enable." Force enable causes all hosts in the cluster to be cryptographically "safe," that is, vCenter Server has installed a host key on the host. See *vSphere Web Services SDK Programming Guide*.

Disk Space Requirements

When you encrypt an existing virtual machine, you need at least twice the space that the virtual machine is currently using.

Encrypted vSphere vMotion

vSphere vMotion always uses encryption when migrating encrypted virtual machines. For virtual machines that are not encrypted, you can select one of the encrypted vSphere vMotion options.

Encrypted vSphere vMotion secures confidentiality, integrity, and authenticity of data that is transferred with vSphere vMotion. vSphere supports encrypted vMotion of unencrypted and encrypted virtual machines across vCenter Server instances.

What Is Encrypted

For encrypted disks, the data is transmitted encrypted in all cases. For unencrypted disks, the following applies:

- If disk data is transferred within a host, that is without changing the host, you change only the datastore, the transfer is unencrypted.
- If disk data is transferred between hosts and encrypted vMotion is used, the transfer is encrypted. If encrypted vMotion is not used the transfer is unencrypted.

For virtual machines that are encrypted, migration with vSphere vMotion always uses encrypted vSphere vMotion. You cannot turn off encrypted vSphere vMotion for encrypted virtual machines.

Encrypted vSphere vMotion States

For virtual machines that are not encrypted, you can set encrypted vSphere vMotion to one of the following states. The default is Opportunistic.

Disabled

Do not use encrypted vSphere vMotion.

Opportunistic

Use encrypted vSphere vMotion if source and destination hosts support it. Only ESXi versions 6.5 and later use encrypted vSphere vMotion.

Required

Allow only encrypted vSphere vMotion. If the source or destination host does not support encrypted vSphere vMotion, migration with vSphere vMotion is not allowed.

When you encrypt a virtual machine, the virtual machine keeps a record of the current encrypted vSphere vMotion setting. If you later disable encryption for the virtual machine, the encrypted vMotion setting remains at Required until you change the setting explicitly. You can change the settings using **Edit Settings**.

See the *vCenter Server and Host Management* documentation for information on enabling and disabling encrypted vSphere vMotion for virtual machines that are not encrypted.

Note Currently, you must use the vSphere APIs to migrate or clone encrypted virtual machines across vCenter Server instances. See *vSphere Web Services SDK Programming Guide* and *vSphere Web Services API Reference*.

Migrating or Cloning Encrypted Virtual Machines Across vCenter Server Instances

vSphere vMotion supports migrating and cloning encrypted virtual machines across vCenter Server instances.

When migrating or cloning encrypted virtual machines across vCenter Server instances, the source and destination vCenter Server instances must be configured to share the key provider that was used to encrypt the virtual machine. In addition, the key provider name must be the same on both the source and destination vCenter Server instances and have the following characteristics:

- Standard key provider: The same key server (or key servers) must be in the key provider.
- Trusted key provider: The same vSphere Trust Authority service must be configured on the destination host.
- vSphere Native Key Provider: Must have the same KDK.

The destination vCenter Server ensures the destination ESXi host has encryption mode enabled, ensuring the host is cryptographically "safe."

The following privileges are required when using vSphere vMotion to migrate or clone an encrypted virtual machine across vCenter Server instances.

- Migrating: **Cryptographic operations.Migrate** on the virtual machine
- Cloning: **Cryptographic operations.Clone** on the virtual machine

Also, the destination vCenter Server must have the **Cryptographic operations.EncryptNew** privilege. If the destination ESXi host is not in "safe" mode, the **Cryptographic operations.RegisterHost** privilege must also be on the destination vCenter Server.

Certain tasks are not allowed when migrating virtual machines (non-encrypted or encrypted), either on the same vCenter Server or across vCenter Server instances.

- You cannot change the VM Storage Policy.
- You cannot perform a key change.

Note You can change the VM Storage Policy while cloning virtual machines.

Minimum Requirements for Migrating or Cloning Encrypted Virtual Machines Across vCenter Server Instances

The minimum version requirements for migrating or cloning standard key provider encrypted virtual machines across vCenter Server instances using vSphere vMotion are:

- Both the source and destination vCenter Server instances must be on version 7.0 or later.
- Both the source and destination ESXi hosts must be on version 6.7 or later.

The minimum version requirements for migrating or cloning trusted key provider encrypted virtual machines across vCenter Server instances using vSphere vMotion are:

- The vSphere Trust Authority service must be configured for the destination host and the destination host must be attested.
- Encryption cannot change on migration. For example, an unencrypted disk cannot be encrypted while the virtual machine is migrated to the new storage.
- You can migrate a standard encrypted virtual machine onto a Trusted Host. The key provider name must be the same on both the source and destination vCenter Server instances.
- You cannot migrate a vSphere Trust Authority encrypted virtual machine onto a non-Trusted Host.

Trusted Key Provider vMotion and Cross-vCenter Server vMotion

Trusted key provider fully supports vMotion across ESXi hosts.

Cross-vCenter Server vMotion is supported, but with the following restrictions.

1. The required trusted service must be configured on the destination host and the destination host must be attested.
2. Encryption cannot change on migration. For example, a disk cannot be encrypted while the virtual machine is migrated to the new storage.

When performing cross-vCenter Server vMotion, vCenter Server checks that the trusted key provider is available on the destination host, and if the host has access to it.

vSphere Native Key Provider vMotion and Cross-vCenter Server vMotion

vSphere Native Key Provider supports vMotion and Encrypted vMotion across ESXi hosts. Cross-vCenter Server vMotion is supported if vSphere Native Key Provider is configured on the destination host.

Encryption Best Practices, Caveats, and Interoperability

Any best practices and caveats that apply to the encryption of physical machines apply to virtual machine encryption as well. The virtual machine encryption architecture results in some

additional recommendations. As you are planning your virtual machine encryption strategy, consider interoperability limitations.

Note For information about vSphere Trust Authority interoperability, see vSphere Trust Authority Best Practices, Caveats, and Interoperability.

Virtual Machine Encryption Best Practices

Follow virtual machine encryption best practices to avoid problems later, for example, when you generate a `vm-support` bundle.

General Best Practices

To avoid problems, follow these general best practices.

- Do not encrypt any vCenter Server appliance virtual machines.
- If your ESXi host fails, retrieve the support bundle as soon as possible. The host key must be available for generating a support bundle that uses a password, or for decrypting a core dump. If the host is rebooted, it is possible that the host key changes. If that happens, you can no longer generate a support bundle with a password or decrypt core dumps in the support bundle with the host key.
- Manage key provider names carefully. If the key provider name changes for a key server that is already in use, a VM that is encrypted with keys from that key server enters a locked state during power-on or register. In that case, remove the key server from the vCenter Server and add it with the key provider name that you used initially.
- Do not edit VMX files and VMDK descriptor files. These files contain the encryption bundle. It is possible that your changes make the virtual machine unrecoverable, and that the recovery problem cannot be fixed.
- The vSphere Virtual Machine Encryption process encrypts data on the host before writing the data to storage. The effectiveness of back-end storage features such as deduplication, compression, replication, and so on, might be affected when encrypting virtual machines in this manner.
- If you use multiple layers of encryption, for example, vSphere Virtual Machine Encryption and in-guest encryption (BitLocker, dm-crypt, and so on), overall virtual machine performance might be affected, because the encryption processes use additional CPU and memory resources.
- Ensure that replicated copies of virtual machines encrypted with vSphere Virtual Machine Encryption have access to the encryption keys at the recovery site. For standard key providers, this is handled as part of the design of the Key Management System, outside of vSphere. For vSphere Native Key Provider, ensure that a backup copy of the Native Key Provider key exists and is protected against loss. For more information, see Back up a vSphere Native Key Provider.

- Encryption is CPU intensive. AES-NI significantly improves encryption performance. Enable AES-NI in your BIOS.

Best Practices for Encrypted Core Dumps

Follow these best practices to avoid having problems when you want to examine a core dump to diagnose a problem.

- Establish a policy regarding core dumps. Core dumps are encrypted because they can contain sensitive information such as keys. If you decrypt a core dump, consider it sensitive information. ESXi core dumps might contain keys for the ESXi host and for the virtual machines on it. Consider changing the host key and recrypting encrypted virtual machines after you decrypt a core dump. You can perform both tasks by using the vSphere API.

  See vSphere Virtual Machine Encryption and Core Dumps for details.

- Always use a password when you collect a `vm-support` bundle. You can specify the password when you generate the support bundle from the vSphere Client or using the `vm-support` command.

  The password recrypts core dumps that use internal keys to use keys that are based on the password. You can later use the password to decrypt any encrypted core dumps that might be included in the support bundle. Unencrypted core dumps and logs are not affected by using the password option.

- The password that you specify during `vm-support` bundle creation is not persisted in vSphere components. You are responsible for keeping track of passwords for support bundles.

- Before you change the host key, generate a `vm-support` bundle with a password. You can later use the password to access any core dumps that might have been encrypted with the old host key.

Key Lifecycle Management Best Practices

Implement best practices that guarantee key server availability and monitor keys on the key server.

- You are responsible for having policies in place that guarantee key server availability.

  If the key server is not available, virtual machine operations that require that vCenter Server request the key from the key server are not possible. That means running virtual machines continue to run, and you can power on, power off, and reconfigure those virtual machines. However, you cannot relocate the virtual machine to a host that does not have the key information.

Most key server solutions include high availability features. You can use the vSphere Client or the API to specify a key provider and the associated key servers.

Note Starting in version 7.0 Update 2, encrypted virtual machines and virtual TPMs can continue to function even when the key server is temporarily offline or unavailable. The ESXi hosts can persist the encryption keys to continue encryption and vTPM operations. See Key Persistence Overview.

- You are responsible for keeping track of keys and for performing remediation if keys for existing virtual machines are not in the Active state.

  The KMIP standard defines the following states for keys.

  - Pre-Active
  - Active
  - Deactivated
  - Compromised
  - Destroyed
  - Destroyed Compromised

  vSphere Virtual Machine Encryption uses only Active keys for encryption. If a key is Pre-Active, vSphere Virtual Machine Encryption activates it. If the key state is Deactivated, Compromised, Destroyed, Destroyed Compromised, you cannot encrypt a virtual machine or disk with that key.

  For keys that are in other states, virtual machines using those keys continue to work. Whether a clone or migration operation succeeds depends on whether they key is already on the host.

  - If the key is on the destination host, the operation succeeds even if the key is not Active on the key server.
  - If the required virtual machine and virtual disk keys are not on the destination host, vCenter Server has to fetch the keys from the key server. If the key state is Deactivated, Compromised, Destroyed, or Destroyed Compromised, vCenter Server displays an error and the operation does not succeed.

  A clone or migration operation succeeds if the key is already on the host. The operation fails if vCenter Server has to pull the keys from the key server.

  If a key is not Active, perform a rekey operation using the API. See the *vSphere Web Services SDK Programming Guide*.

- Develop key rotation policies so that keys are retired and rolled over after a specific time.
  - Trusted key provider: Change the primary key of a trusted key provider.
  - vSphere Native Key Provider: Change the `key_id` of a vSphere Native Key Provider.

Backup and Restore Best Practices

Set up policies on backup and restore operations.

- Not all backup architectures are supported. See Virtual Machine Encryption Interoperability.
- Set up policies for restore operations. Because backup is always in cleartext, plan to encrypt virtual machines right after restore is finished. You can specify that the virtual machine is encrypted as part of the restore operation. If possible, encrypt virtual machine as part of the restore process to avoid exposing sensitive information. To change the encryption policy for any disks that are associated with the virtual machine, change the storage policy for the disk.
- Because the VM home files are encrypted, ensure that the encryption keys are available at the time of a restore.

Performance Best Practices

- Encryption performance depends on the CPU and storage speed.
- Encrypting existing virtual machines is more time consuming than encrypting a virtual machine during creation. Encrypt a virtual machine when you create it if possible.

Storage Policy Best Practices

Do not modify the bundled VM Encryption sample storage policy. Instead, clone the policy and edit the clone.

Note No automated way of returning VM Encryption Policy to its original settings exists.

See the *vSphere Storage* documentation for details customizing storage policies.

Removing Encryption Keys Best Practices

To ensure that encryption keys are removed from a cluster, after deleting, unregistering, or moving the encrypted virtual machine to another vCenter Server, reboot the ESXi hosts in the cluster.

Virtual Machine Encryption Caveats

Review Virtual Machine Encryption caveats to avoid problems later.

To understand which devices and features cannot be used with Virtual Machine Encryption, see Virtual Machine Encryption Interoperability.

Limitations

Consider the following caveats when you plan your virtual machine encryption strategy.

- When you clone an encrypted virtual machine or perform a Storage vMotion operation, you can attempt to change the disk format. Such conversions do not always succeed. For example, if you clone a virtual machine and attempt to change the disk format from lazy-zeroed thick format to thin format, the virtual machine disk keeps the lazy-zeroed thick format.

- When you detach a disk from a virtual machine, the storage policy information for the virtual disk is not retained.
  - If the virtual disk is encrypted, you must explicitly set the storage policy to VM Encryption Policy or to a storage policy that includes encryption.
  - If the virtual disk is not encrypted, you can change the storage policy when you add the disk to a virtual machine.

  See Virtual Disk Encryption for details.
- Decrypt core dumps before moving a virtual machine to a different cluster.

  The vCenter Server does not store KMS keys but only tracks the key IDs. As a result, vCenter Server does not store the ESXi host key persistently.

  Under certain circumstances, for example, when you move the ESXi host to a different cluster and reboot the host, vCenter Server assigns a new host key to the host. You cannot decrypt any existing core dumps with the new host key.
- OVF Export is not supported for an encrypted virtual machine.
- Using the VMware Host Client to register an encrypted virtual machine is not supported.

Virtual Machine Locked State

If the virtual machine key or one or more of the virtual disk keys are missing, the virtual machine enters a locked state. In a locked state, you cannot perform virtual machine operations.

- When you encrypt both a virtual machine and its disks from the vSphere Client, the same key is used for both.
- When you perform the encryption using the API, you can use different encryption keys for the virtual machine and for disks. In that case, if you attempt to power on a virtual machine, and one of the disk keys is missing, the power on operation fails. If you remove the virtual disk, you can power on the virtual machine.

See Resolve Missing Key Issues for troubleshooting suggestions.

Virtual Machine Encryption Interoperability

vSphere Virtual Machine Encryption has some limitations regarding devices and features that it can interoperate with.

The following limitations and remarks refer to using vSphere Virtual Machine Encryption. For similar information about using vSAN encryption, see the *Administering VMware vSAN* documentation.

Limitations on Certain Encryption Tasks

Some restrictions apply when performing certain tasks on an encrypted virtual machine.

- For most virtual machine encrypted operations, you must power off the virtual machine. You can clone an encrypted virtual machine and you can perform a shallow recrypt while the virtual machine is powered on.

  Note Virtual machines configured with IDE controllers must be powered off to perform a shallow rekey operation.

- You cannot perform a deep recrypt on a virtual machine with snapshots. You can perform a shallow recrypt on a virtual machine with snapshots.

Virtual Trusted Platform Module Devices and vSphere Virtual Machine Encryption

A virtual Trusted Platform Module (vTPM) is a software-based representation of a physical Trusted Platform Module 2.0 chip. You can add a vTPM to either a new or an existing virtual machine. To add a vTPM to a virtual machine, you must configure a key provider in your vSphere environment. When you configure a vTPM, the virtual machine “home” files are encrypted (memory swap, NVRAM files, and so on). The disk files, or VMDK files, are not automatically encrypted. You can choose to add encryption explicitly for the virtual machine disks.

Caution Cloning a virtual machine duplicates the entire virtual machine, including the virtual devices such as a vTPM. Information stored in the vTPM, including properties of the vTPM that software can use to determine a system’s identity, is also duplicated.

vSphere Virtual Machine Encryption and Suspended State and Snapshots

You can resume from a suspended state of an encrypted virtual machine, or revert to a memory snapshot of an encrypted machine. You can migrate an encrypted virtual machine with memory snapshot and suspended state between ESXi hosts.

vSphere Virtual Machine Encryption and IPv6

You can use vSphere Virtual Machine Encryption with pure IPv6 mode or in mixed mode. You can configure the key server with IPv6 addresses. You can configure both the vCenter Server and the key server with only IPv6 addresses.

Limitations on Cloning in vSphere Virtual Machine Encryption

Certain cloning features do not work with vSphere Virtual Machine Encryption.

- For a standard key provider, cloning is supported conditionally.
  - Full clones are supported. The clone inherits the parent encryption state including keys. You can encrypt the full clone, re-encrypt the full clone to use new keys, or decrypt the full clone.

Linked clones are supported and the clone inherits the parent encryption state including keys. You cannot decrypt the linked clone or re-encrypt a linked clone with different keys.

Note Verify that other applications support linked clones. For example, VMware Horizon® 7 supports both full clones and instant clones, but not linked clones.

- For a trusted key provider or a vSphere Native Key Provider, cloning is supported, but encryption keys cannot be changed on clone. This behavior contrasts with standard encryption where you can change keys when creating a clone. The following operations are not supported by vSphere Trust Authority or vSphere Native Key Provider during cloning of a virtual machine:
  - Cloning from an unencrypted virtual machine to an encrypted virtual machine
  - Cloning from an encrypted virtual machine and changing the encryption keys
  - Cloning from an encrypted virtual machine to an unencrypted virtual machine
- Instant clone is supported by all key provider types, but you cannot change encryption keys on clone.

Unsupported Disk Configurations with vSphere Virtual Machine Encryption

Certain types of virtual machine disk configurations are not supported with vSphere Virtual Machine Encryption.

- RDM (Raw Device Mapping). However, vSphere Virtual Volumes (vVols) are supported.
- Multi-writer or shared disks (MSCS, WSFC, or Oracle RAC). Encrypted virtual machine "home" files are supported for multi-writer disks. Encrypted virtual disks are not supported for multi-writer disks. If you attempt to select Multi-writer in the **Edit Settings** page of the virtual machine with encrypted virtual disks, the **OK** button is deactivated.

Miscellaneous Limitations in vSphere Virtual Machine Encryption

Other features that do not work with vSphere Virtual Machine Encryption include the following:

- vSphere ESXi Dump Collector
- Content Library
  - Content libraries support two types of templates, the OVF Template type and the VM Template type. You cannot export an encrypted virtual machine to the OVF Template type. The OVF Tool does not support encrypted virtual machines. You can create encrypted VM templates using the VM Template type. See the *vSphere Virtual Machine Administration* documentation.

- Software for backing up encrypted virtual disks must use the VMware vSphere Storage API - Data Protection (VADP) to either back up the disks in hot-add mode or NBD mode with SSL enabled. However, not all backup solutions that use VADP for virtual disk backup are supported. Check with your backup vendor for details.
  - VADP SAN transport mode solutions are not supported for backing up encrypted virtual disks.
  - VADP Hot-Add solutions are supported for encrypted virtual disks. The backup software must support encryption of the proxy VM that is used as part of the hot-add backup workflow. The vendor must have the privilege **Cryptographic Operations.Encrypt Virtual Machine**.
  - Backup solutions using the NBD-SSL transport modes are supported for backing up encrypted virtual disks. The vendor application must have the privilege **Cryptographic Operations.Direct Access**.
- You cannot send output from an encrypted virtual machine to a serial port or parallel port. Even if the configuration appears to succeed, output is sent to a file.
- vSphere Virtual Machine Encryption is not supported in VMware Cloud on AWS. See the *Managing the VMware Cloud on AWS Data Center* documentation.

Key Persistence Overview

In vSphere 7.0 Update 2 and later, encrypted virtual machines and virtual TPMs can continue to optionally function even when the key server is temporarily offline or unavailable. The ESXi hosts can persist the encryption keys to continue encryption and vTPM operations.

Before vSphere 7.0 Update 2, encrypted virtual machines and vTPMs require that the key server always is available to function. In vSphere 7.0 Update 2 and later, encrypted devices can function even when access to a key server is disrupted.

Starting in vSphere 7.0 Update 3, encrypted vSAN clusters can also function even when access to a key provider is disrupted.

Note Key persistence is not necessary when using vSphere Native Key Provider. vSphere Native Key Provider is designed out-of-the-box to run without requiring access to a key server. See the following section, "Key Persistence and vSphere Native Key Provider."

Key Persistence on the ESXi Host

When using a standard key provider, the ESXi host relies on vCenter Server to manage the encryption keys. When using a trusted key provider, the ESXi host relies directly on the Trust Authority Hosts for keys, and vCenter Server is not involved.

Regardless of the type of key provider, the ESXi host obtains the keys initially and retains them in its key cache. If the ESXi host reboots, it loses its key cache. The ESXi host then requests the keys again, either from the key server (standard key provider), or the Trust Authority Hosts (trusted key provider). When the ESXi host tries to obtain keys and the key server is offline or unreachable, vTPMs and workload encryption cannot function. For edge-style deployments, in which a key server is typically not deployed on site, a loss of connectivity to a key server can cause unnecessary downtime for encrypted workloads.

In vSphere 7.0 Update 2 and later, encrypted workloads can continue to function even when the key server is offline or unreachable. If the ESXi host has a TPM, the encryption keys are persisted in the TPM across reboots. So, even if an ESXi host reboots, the host does not need to request encryption keys. Also, encryption and decryption operations can continue when the key server is unavailable, because the keys have persisted in the TPM. In essence, when either the key server or Trust Authority Hosts are unavailable, you can keep running encrypted workloads "key server free." Also, vTPMs can likewise continue to function even when the key server is unreachable.

Key Persistence and vSphere Native Key Provider

When using a vSphere Native Key Provider, vSphere generates encryption keys and no key server is required. The ESXi hosts get a Key Derivation Key (KDK), which is used to derive other keys. After receiving the KDK and generating other keys, the ESXi hosts do not need access to vCenter Server to do encryption operations. In essence, a vSphere Native Key Provider always runs "key server free."

The KDK persists on an ESXi host by default even after reboot, and even when vCenter Server is not available after the host reboots.

You can activate key persistence with vSphere Native Key Provider, but it is normally unnecessary. The ESXi hosts have complete access to vSphere Native Key Provider and so additional persistence of keys is redundant. One use case for activating key persistence with vSphere Native Key Provider is when you also have a standard key provider (external KMIP server) configured.

How Do You Set Up Key Persistence

To enable or disable key persistence, see Enable and Disable Key Persistence on an ESXi Host.

Configuring and Managing a Standard Key Provider

7

Using a standard key provider in your vSphere environment requires some preparation. After your environment is set up, you can create encrypted virtual machines and virtual disks and encrypt existing virtual machines and disks.

After your environment is set up for a standard key provider, you can use the vSphere Client to create encrypted virtual machines and virtual disks and encrypt existing virtual machines and disks. See Chapter 10 Use Encryption in Your vSphere Environment.

You can perform additional tasks by using the API and by using the crypto-util CLI. See the *vSphere Web Services SDK Programming Guide* for the API documentation and the `crypto-util` command-line help for details about that tool.

Read the following topics next:

- Standard Key Provider Overview
- Set Up the Standard Key Provider
- Set Up Separate Key Providers for Different Users
- Delete a Standard Key Provider

Standard Key Provider Overview

You can use a standard key provider to perform virtual machine encryption tasks.

What Is a Standard Key Provider?

In vSphere, a standard key provider gets encryption keys directly from a key server, and the vCenter Server distributes the keys to the required ESXi hosts in a data center.

You can add separate standard key providers for different users and set the default standard key provider.

vSphere Standard Key Provider Requirements

- vSphere 6.5 or later
- An external key server (KMS)

The Key Management Server must support the Key Management Interoperability Protocol (KMIP) 1.1 standard. See the *vSphere Compatibility Matrices* for details.

You can find information about VMware certified KMS vendors in the VMware Compatibility Guide under Platform and Compute. If you select Compatibility Guides, you can open the Key Management Server (KMS) compatibility documentation. This documentation is updated frequently.

Standard Key Provider Privileges

Standard key providers use the **Cryptographer.*** privileges. See Cryptographic Operations Privileges.

Set Up the Standard Key Provider

Before you can start with virtual machine encryption tasks, you must set up the standard key provider.

Setting up a standard key provider includes adding the key provider and establishing trust with the key server. When you add a key provider, you are prompted to make it the default. You can explicitly change the default key provider. vCenter Server provisions keys from the default key provider.

Note What was previously called a Key Management Server cluster in vSphere 6.5 and 6.7 is now called a key provider.

 (Virtual Machine Encryption Setting Up a Standard Key Provider)

Add a Standard Key Provider Using the vSphere Client

You can add a standard key provider to your vCenter Server system from the vSphere Client or by using the public API.

The vSphere Client enables you to add a standard key provider to your vCenter Server system, and establish trust between the key server and vCenter Server.

- You can add multiple key servers from the same vendor.
- If your environment supports solutions from different vendors, you can add multiple key providers.
- If your environment includes multiple key providers, and you delete the default key provider, you must set another default explicitly.
- You can configure the key server with IPv6 addresses.
  - Both the vCenter Server system and the key server can be configured with only IPv6 addresses.

Prerequisites

- Verify that the key server (KMS) is in the *VMware Compatibility Guide for Key Management Servers (KMS)* and is KMIP 1.1 compliant, and that it can be a symmetric key foundry and server.
- Verify that you have the required privileges: **Cryptographic operations.Manage key servers**.
- Ensure that the key server is highly available. Loss of connection to the key server, such as during a power outage or a disaster recovery event, renders encrypted virtual machines inaccessible.

  Note Starting in vSphere 7.0 Update 2, encrypted virtual machines and virtual TPMs can continue to function even when the key server is temporarily offline or unavailable. See Key Persistence Overview.

- Consider your infrastructure's dependencies on the key server carefully. Some KMS solutions are delivered as virtual appliances, making it possible to create a dependency loop or other availability problem with poor placement of the KMS appliance.

Procedure

1. Log in to the vCenter Server system with the vSphere Client.
2. Browse the inventory list and select the vCenter Server instance.
3. Click **Configure**, and under **Security** click **Key Providers**.
4. Click **Add Standard Key Provider** and enter the key provider information.

Option	Value
Name	Name for the key provider. Each logical key provider, regardless of its type (Standard, Trusted, and Native Key Provider), must have a unique name across all vCenter Server systems. For more information, see Key Provider Naming.
KMS	Alias for the key server (KMS).
Address	IP address or FQDN of the key server.
Port	Port on which vCenter Server connects to the key server.
Proxy server	Optional proxy server address for connecting to the key server.
Proxy port	Optional proxy port for connecting to the key server.
Username	Some key server vendors allow users to isolate encryption keys that are used by different users or groups by specifying a user name and password. Specify a user name only if your key server supports this functionality, and if you intend to use it.
Password	Some key server vendors allow users to isolate encryption keys that are used by different users or groups by specifying a user name and password. Specify a password only if your key server supports this functionality, and if you intend to use it.

You can click **Add KMS** to add more key servers.

5 Click **Add Key Provider**.

6 Click **Trust**.

vCenter Server adds the key provider and displays the status as Connected.

What to do next

See Establish a Standard Key Provider Trusted Connection by Exchanging Certificates.

Establish a Standard Key Provider Trusted Connection by Exchanging Certificates

After you add the standard key provider to the vCenter Server system, you can establish a trusted connection. The exact process depends on the certificates that the key provider accepts, and on your company policy.

Prerequisites

Add the standard key provider.

Procedure

1 Navigate to the vCenter Server.

2 Click **Configure** and select **Key Providers** under **Security**.

3 Select the key provider.

The KMS for the key provider is displayed.

4 Select the KMS.

5 From the **Establish Trust** drop-down menu, select **Make KMS trust vCenter**.

6 Select the option appropriate for your server and follow the steps.

Option	See
vCenter Server Root CA certificate	Use the Root CA Certificate Option to Establish a Standard Key Provider Trusted Connection.
vCenter Server Certificate	Use the Certificate Option to Establish a Standard Key Provider Trusted Connection.
Upload certificate and private key	Use the Upload Certificate and Private Key Option to Establish a Standard Key Provider Trusted Connection.
New Certificate Signing Request	Use the New Certificate Signing Request Option to Establish a Standard Key Provider Trusted Connection.

Use the Root CA Certificate Option to Establish a Standard Key Provider Trusted Connection

Some Key Management Server (KMS) vendors require that you upload your root CA certificate to the KMS. All certificates that are signed by your root CA are then trusted by this KMS.

The root CA certificate that vSphere Virtual Machine Encryption uses is a self-signed certificate that is stored in a separate store in the VMware Endpoint Certificate Store (VECS) on the vCenter Server system.

Note Generate a root CA certificate only if you want to replace existing certificates. If you do, other certificates that are signed by that root CA become invalid. You can generate a new root CA certificate as part of this workflow.

Procedure

1. Navigate to the vCenter Server.
2. Click **Configure** and select **Key Providers** under **Security**.
3. Select the key provider with which you want to establish a trusted connection.

   The KMS for the key provider is displayed.
4. From the **Establish Trust** drop-down menu, select **Make KMS trust vCenter**.
5. Select **vCenter Root CA Certificate** and click **Next**.

   The Download Root CA Certificate dialog box is populated with the root certificate that vCenter Server uses for encryption. This certificate is stored in VECS.
6. Copy the certificate to the clipboard or download the certificate as a file.
7. Follow the instructions from your KMS vendor to upload the certificate to their system.

   Note Some KMS vendors require that the KMS vendor restarts the KMS to pick up the root certificate that you upload.

What to do next

Finalize the certificate exchange. See Finish the Trust Setup for a Standard Key Provider.

Use the Certificate Option to Establish a Standard Key Provider Trusted Connection

Some Key Management Server (KMS) vendors require that you upload the vCenter Server certificate to the KMS. After the upload, the KMS accepts traffic that comes from a system with that certificate.

vCenter Server generates a certificate to protect connections with the KMS. The certificate is stored in a separate key store in the VMware Endpoint Certificate Store (VECS) on the vCenter Server system.

Procedure

1. Navigate to the vCenter Server.
2. Click **Configure** and select **Key Providers** under **Security**.
3. Select the key provider with which you want to establish a trusted connection.

   The KMS for the key provider is displayed.
4. From the **Establish Trust** drop-down menu, select **Make KMS trust vCenter**.
5. Select **vCenter Certificate** and click **Next**.

   The Download Certificate dialog box is populated with the root certificate that vCenter Server uses for encryption. This certificate is stored in VECS.

   Note Do not generate a new certificate unless you want to replace existing certificates.
6. Copy the certificate to the clipboard or download it as a file.
7. Follow the instructions from your KMS vendor to upload the certificate to the KMS.

What to do next

Finalize the trust relationship. See Finish the Trust Setup for a Standard Key Provider.

Use the Upload Certificate and Private Key Option to Establish a Standard Key Provider Trusted Connection

Some Key Management Server (KMS) vendors require that you upload the KMS server certificate and private key to the vCenter Server system.

Some KMS vendors generate a certificate and private key for the connection and make them available to you. After you upload the files, the KMS trusts your vCenter Server instance.

Prerequisites

- Request a certificate and private key from the KMS vendor. The files are X509 files in PEM format.

Procedure

1. Navigate to the vCenter Server.
2. Click **Configure** and select **Key Providers** under **Security**.
3. Select the key provider with which you want to establish a trusted connection.

   The KMS for the key provider is displayed.
4. From the **Establish Trust** drop-down menu, select **Make KMS trust vCenter**.
5. Select **KMS certificate and private key** and click **Next**.
6. Paste the certificate that you received from the KMS vendor into the top text box or click **Upload a File** to upload the certificate file.

7 Paste the key file into the bottom text box or click **Upload a File** to upload the key file.

8 Click **Establish Trust**.

What to do next

Finalize the trust relationship. See Finish the Trust Setup for a Standard Key Provider.

Use the New Certificate Signing Request Option to Establish a Standard Key Provider Trusted Connection

Some Key Management Server (KMS) vendors require that vCenter Server generate a Certificate Signing Request (CSR) and send that CSR to the KMS. The KMS signs the CSR and returns the signed certificate. You can upload the signed certificate to vCenter Server.

Using the **New Certificate Signing Request** option is a two-step process. First you generate the CSR and send it to the KMS vendor. Then you upload the signed certificate that you receive from the KMS vendor to vCenter Server.

Procedure

1 Navigate to the vCenter Server.

2 Click **Configure** and select **Key Providers** under **Security**.

3 Select the key provider with which you want to establish a trusted connection.

The KMS for the key provider is displayed.

4 From the **Establish Trust** drop-down menu, select **Make KMS trust vCenter**.

5 Select **New Certificate Signing Request (CSR)** and click **Next**.

6 In the dialog box, copy the full certificate in the text box to the clipboard or download it as a file.

Use the **Generate new CSR** button in the dialog box only if you explicitly want to generate a CSR. Using that option makes any signed certificates that are based on the old CSR invalid.

7 Follow the instructions from your KMS vendor to submit the CSR.

8 When you receive the signed certificate from the KMS vendor, click **Key Providers** again, select the key provider, and from the **Establish Trust** drop-down menu, select **Upload Signed CSR Certificate**.

9 Paste the signed certificate into the bottom text box or click **Upload File** and upload the file, and click **Upload**.

What to do next

Finalize the trust relationship. See Finish the Trust Setup for a Standard Key Provider.

Set the Default Key Provider

You must set the default key provider if you do not make the first key provider the default, or if your environment uses multiple key providers and you remove the default one.

Prerequisites

As a best practice, verify that the Connection Status in the **Key Providers** tab shows Connected and a green check mark.

Procedure

1. Navigate to the vCenter Server.
2. Click **Configure** and select **Key Providers** under **Security**.
3. Select the key provider.
4. Click **Make Default**.

   A confirmation dialog box appears.
5. Click **Make Default**.

   The key provider displays as the current default.

Finish the Trust Setup for a Standard Key Provider

Unless the **Add Standard Key Provider** dialog box prompted you to trust the KMS, you must explicitly establish trust after certificate exchange is complete.

You can complete the trust setup, that is, make vCenter Server trust the KMS, either by trusting the KMS or by uploading a KMS certificate. You have two options:

- Trust the certificate explicitly by using the **Upload KMS certificate** option.
- Upload a KMS leaf certificate or the KMS CA certificate to vCenter Server by using the **Make vCenter Trust KMS** option.

Note If you upload the root CA certificate or the intermediate CA certificate, vCenter Server trusts all certificates that are signed by that CA. For strong security, upload a leaf certificate or an intermediate CA certificate that the KMS vendor controls.

Procedure

1. Navigate to the vCenter Server.
2. Click **Configure** and select **Key Providers** under **Security**.
3. Select the key provider with which you want to establish a trusted connection.

   The KMS for the key provider is displayed.
4. Select the KMS.

5 Select one of the following options from the **Establish Trust** drop-down menu.

Option	Action
Make vCenter Trust KMS	In the dialog box that appears, click **Trust**.
Upload KMS certificate	a In the dialog box that appears, either paste in the certificate, or click **Upload a file** and browse to the certificate file. b Click **Upload**.

Set Up Separate Key Providers for Different Users

You can set up your environment with different key providers for different users of the same KMS instance. Having multiple key providers is helpful, for example, if you want to grant different departments in your company access to different sets of encryption keys.

You can use multiple key providers for the same KMS to separate keys. Having separate sets of keys is essential for use cases like different BUs or different customers.

Note Not all KMS vendors support multiple users.

Prerequisites

Set up the connection with the KMS.

Procedure

1 Create two users with corresponding user names and passwords, for example C1 and C2, on the KMS.

2 Log in to vCenter Server and create the first key provider.

3 When prompted for a user name and password, give information that is unique to the first user.

4 Create a second key provider and add the same KMS, but use the second user name and password (C2).

Results

The two key providers have independent connections to the KMS and use a different set of keys.

Delete a Standard Key Provider

You can use the vSphere Client to delete a standard key provider from vCenter Server.

After you delete a standard key provider, virtual machines that have vTPMs or that are encrypted continue to run. If you reboot the ESXi host, its encrypted virtual machines enter a locked state. After you unregister these virtual machines, they enter a locked state when you try to re-register them. The only way to unlock the virtual machines is to restore the previous standard key provider.

Prerequisites

Required privilege: **Cryptographic operations.Manage key servers**

Before you delete a standard key provider, rekey any encrypted virtual machines and datastores that were encrypted using that key provider to another key provider. See Rekey an Encrypted Virtual Machine Using the vSphere Client.

In addition, maintain a backup of the standard key provider in case you must rekey an encrypted virtual machine after deleting the key provider.

Procedure

1. Log in to the vCenter Server system with the vSphere Client.
2. Browse the inventory list and select the vCenter Server instance.
3. Click **Configure**, and under **Security** click **Key Providers**.
4. Select the standard key provider you want to delete.
5. Click **Delete**.
6. Read the warning message and slide the slider all the way to the right.
7. Click **Delete**.

Results

The standard key provider is removed from the vCenter Server.

Configuring and Managing vSphere Native Key Provider

8

Using a VMware vSphere® Native Key Provider™ in your vSphere environment requires some preparation. After you configure vSphere Native Key Provider, you can create virtual Trusted Platform Modules (vTPMs) on your virtual machines.

After you set up your environment for vSphere Native Key Provider, you can use the vSphere Client and API to create vTPMs. If you purchase the VMware vSphere® Enterprise Plus Edition™, you can also encrypt virtual machines and virtual disks, and encrypt existing virtual machines and disks.

 (Configuring a vSphere Native Key Provider)

Read the following topics next:

- vSphere Native Key Provider Overview
- vSphere Native Key Provider Process Flows
- Configure a vSphere Native Key Provider
- Back up a vSphere Native Key Provider
- Recovering a vSphere Native Key Provider
- Update a vSphere Native Key Provider
- Delete a vSphere Native Key Provider

vSphere Native Key Provider Overview

In vSphere 7.0 Update 2 and later, you can use the built-in vSphere Native Key Provider to enable encryption technologies, such as virtual TPMs (vTPM).

vSphere Native Key Provider is included in all vSphere editions and does not require an external key server (also called a Key Management Server (KMS) in the industry). You can also use vSphere Native Key Provider for vSphere Virtual Machine Encryption, but you must purchase the VMware vSphere® Enterprise Plus Edition™.

What Is vSphere Native Key Provider

With a standard key provider or trusted key provider, you must configure an external key server. In a standard key provider setup, vCenter Server fetches the keys from the external key server and distributes them to the ESXi hosts. In a trusted key provider (vSphere Trust Authority) setup, the trusted ESXi hosts fetch the keys directly.

With vSphere Native Key Provider, you no longer need an external key server. vCenter Server generates a primary key, called the Key Derivation Key (KDK), and pushes it to all ESXi hosts in the cluster. The ESXi hosts then generate data encryption keys (even when not connected to vCenter Server) to enable security functionality such as vTPMs. vTPM functionality is included in all vSphere editions. To use vSphere Native Key Provider for vSphere Virtual Machine Encryption, you must have purchased the vSphere Enterprise Plus Edition. vSphere Native Key Provider can coexist with an existing key server infrastructure.

vSphere Native Key Provider:

- Enables the use of vTPMs, vSphere Virtual Machine Encryption, and vSAN Data at Rest Encryption, when you do not require or want an external key server.
- Works only with VMware infrastructure products.
- Does not provide external interoperability, KMIP support, hardware security modules, or other features that a traditional, third-party, external key server can offer for interoperability or regulatory compliance. If your organization requires this functionality for non-VMware products and components, install a traditional, third-party key server.
- Helps address the needs of organizations that either cannot use, or do not want not to use, an external key server.
- Improves data sanitization and system reuse practices by enabling earlier use of encryption technologies on media that is difficult to sanitize, such as flash and SSD.
- Provides a transition path between key providers. vSphere Native Key Provider is compatible with the VMware standard key provider and the vSphere Trust Authority trusted key provider.
- Works with multiple vCenter Server systems using an Enhanced Linked Mode configuration or a vCenter Server High Availability configuration.
- Can be used to enable vTPM in all editions of vSphere, and encrypt virtual machines with the purchase of the vSphere Enterprise Plus Edition that includes vSphere Virtual Machine Encryption. vSphere Virtual Machine Encryption works with vSphere Native Key Provider as it does with VMware standard and trusted key providers.
- Can be used to enable vSAN Data at Rest Encryption with the use of an appropriate vSAN license.

- Can use a Trusted Platform Module (TPM) 2.0 to increase security when one is installed in an ESXi host. You can also configure vSphere Native Key Provider to be available only to hosts where a TPM 2.0 is installed. If you use a TPM, it must be TPM 2.0. vSphere Native Key Provider does not support TPM 1.2.

Note An ESXi host does not require a TPM 2.0 to use a vSphere Native Key Provider. However, a TPM 2.0 does provide enhanced security.

As with all security solutions, consider the system design, implementation considerations, and tradeoffs of using Native Key Provider. For example, ESXi key persistence avoids the dependency on a key server always being available. However, because key persistence stores the Native Key Provider cryptographic information on the clustered hosts, you still are at risk if malicious actors steal the ESXi hosts themselves. Because environments differ, assess and implement your security controls in accordance with your organization's regulatory and security needs, operational requirements, and tolerance for risk.

For more overview information about vSphere Native Key Provider, see https://core.vmware.com/native-key-provider.

vSphere Native Key Provider Requirements

To use vSphere Native Key Provider, you must:

- Ensure that both the vCenter Server system and ESXi hosts are running vSphere 7.0 Update 2 or later.
- Configure the ESXi hosts in a cluster. While not required, as a best practice, use ESXi hosts that are as identical as possible, including TPMs. Cluster management and feature enablement is much easier when cluster hosts are identical.
- Configure the vCenter Server file-based backup and restore, and store the backups securely as they contain the Key Derivation Key. See the topic on vCenter Server backup and restore in the *vCenter Server Installation and Setup* documentation.

To perform vSphere Virtual Machine Encryption or vSAN encryption using vSphere Native Key Provider, you must purchase the edition of those products containing the appropriate license.

vSphere Native Key Provider and Enhanced Linked Mode

You can configure a single vSphere Native Key Provider that is shareable across vCenter Server systems configured in an Enhanced Linked Mode configuration. The high-level steps in this scenario are:

1. Creating the vSphere Native Key Provider on one of the vCenter Server systems
2. Backing up the Native Key Provider on the vCenter Server on which it was created
3. Exporting the Native Key Provider
4. Restoring the Native Key Provider to the other vCenter Server systems in the Enhanced Link Mode configuration (see Restore a vSphere Native Key Provider Using the vSphere Client)

vSphere Native Key Provider Privileges

As with standard and trusted key providers, vSphere Native Key Provider uses the **Cryptographer.*** privileges. In addition, vSphere Native Key Provider uses the **Cryptographer.ReadKeyServersInfo** privilege, which is specific to vSphere Native Key Providers, to list vSphere Native Key Providers. See Cryptographic Operations Privileges.

vSphere Native Key Provider Alarms

You must back up a vSphere Native Key Provider. When a vSphere Native Key Provider is not backed up, vCenter Server generates an alarm. When you back up the vSphere Native Key Provider for which an alarm was generated, vCenter Server resets the alarm. By default, vCenter Server checks for backed-up vSphere Native Key Providers once a day. You can change the checking interval by modifying the `vpxd.KMS.backupCheckInterval` option.

vSphere Native Key Provider Periodic Remediation Check

vCenter Server checks periodically that the vSphere Native Key Provider configuration on vCenter Server and ESXi hosts matches. When a host state changes, for example, when you add a host to the cluster, the key provider configuration on the cluster drifts away from the configuration on the host. If the configuration (keyID) differs on the host, vCenter Server updates the configuration of the host automatically. No manual intervention is required.

By default, vCenter Server checks the configuration every five minutes. You can modify the interval by using the `vpxd.KMS.remediationInterval` option.

Using vSphere Native Key Provider with a Disaster Recovery Site

You can use vSphere Native Key Provider with a backup disaster recovery site. Importing the vSphere Native Key Provider backup from the primary vCenter Server to the vCenter Server backup at the disaster recovery site enables that cluster to decrypt and run your encrypted virtual machines.

Always test your DR solution. Never assume that your solution works without trying a recovery. Ensure that a copy of the vSphere Native Key Provider backup is also available to your DR site.

Unsupported Features in vSphere Native Key Provider

Currently, vSphere Native Key Provider does not support the following:

- First Class Disk (FCD) encryption

vSphere Native Key Provider Process Flows

Understanding vSphere Native Key Provider process flows is essential to learning how to configure and manage your vSphere Native Key Provider.

You can use the built-in vSphere Native Key Provider to power encryption-based virtual TPMs (vTPM). vSphere Native Key Provider is included in all vSphere editions and does not require an external key server (KMS). To use vSphere Native Key Provider for vSphere Virtual Machine Encryption, you must purchase the vSphere Enterprise+ edition.

Configuring vSphere Native Key Provider

Configuring vSphere Native Key Provider involves these basic operations:

1. A user with the appropriate administrative privileges uses the vSphere Client to create a vSphere Native Key Provider on a vCenter Server.
2. The vCenter Server then configures the vSphere Native Key Provider for all clusters of ESXi hosts.

   In this step, vCenter Server pushes a primary key to all ESXi hosts in the cluster. Likewise, if you update or delete a vSphere Native Key Provider, the change is pushed to the hosts in the cluster.
3. Users with the appropriate cryptographic privileges create vTPMs and encrypted virtual machines (provided you have purchased the vSphere Enterprise+ edition).

   See Chapter 11 Securing Virtual Machines with Virtual Trusted Platform Module and Chapter 10 Use Encryption in Your vSphere Environment.

vSphere Native Key Provider Encryption Process Flow

To understand how different components interact to perform an encryption task using vSphere Native Key Provider, see Encryption Process Flow.

Configure a vSphere Native Key Provider

Before you can start with encryption tasks, you must configure a vSphere Native Key Provider on vCenter Server.

vSphere 7.0 Update 2 and later includes a key provider called vSphere Native Key Provider. vSphere Native Key Provider enables encryption-related functionality without requiring an external key server (KMS). Initially, vCenter Server is not configured with a vSphere Native Key Provider. You must manually configure a vSphere Native Key Provider.

An ESXi host does not require a TPM 2.0 to use a vSphere Native Key Provider. However, a TPM 2.0 does provide enhanced security.

Note When you configure vSphere Native Key Provider, the key providers are available on all clusters for the vCenter Server on which you configure them. As a result, all hosts attached to the vCenter Server have access to all the vSphere Native Key Providers that you configure.

Prerequisites

Required privilege: **Cryptographic operations.Manage key servers**

Procedure

1. Log in to the vCenter Server system with the vSphere Client.
2. Browse the inventory list and select the vCenter Server instance.
3. Click **Configure**, and under **Security** click **Key Providers**.
4. Click **Add** then click **Add Native Key Provider**.
5. Enter a name for the vSphere Native Key Provider.

   Each logical key provider, regardless of its type (Standard, Trusted, and Native Key Provider), must have a unique name across all vCenter Server systems.

   For more information, see Key Provider Naming.
6. If you want this vSphere Native Key Provider to be used only by hosts with a TPM 2.0, select the **Use key provider only with TPM protected ESXi hosts** check box.

   If enabled, the vSphere Native Key Provider is available only on hosts with a TPM 2.0.
7. Click **Add Key Provider**.

   Note It takes about five minutes for all the clustered ESXi hosts in a data center to get the key provider, and for the vCenter Server to update its cache. Because of the way the information is propagated, you might have to wait for a few minutes to use the key provider for key operations on some of the hosts.

Results

The vSphere Native Key Provider is added and appears in the **Key Provider** pane. At this point, the vSphere Native Key Provider is not backed up. You must back up the vSphere Native Key Provider before you can use it.

What to do next

See Back up a vSphere Native Key Provider.

Back up a vSphere Native Key Provider

In case you must restore the key provider configuration, backing up a vSphere Native Key Provider is required as part of a disaster recovery scenario. You can use the vSphere Client, PowerCLI, or API to back up the vSphere Native Key Provider.

vSphere Native Key Provider is backed up as part of the vCenter Server file-based backup. However, you must back up the vSphere Native Key Provider at least once before you can use it. When you create a vSphere Native Key Provider, it is not backed up.

A backup is necessary in case you must restore the configuration. To restore a vSphere Native Key Provider, see Restore a vSphere Native Key Provider Using the vSphere Client.

Keep the backup file in a secure location. You can password-protect the backup when you create it. The backup file is in PKCS#12 format.

vCenter Server creates an alarm if a vSphere Native Key Provider has not been backed up. You can acknowledge the alarm, but it reappears every 24 hours until you have backed up the vSphere Native Key Provider.

Prerequisites

Required privilege: **Cryptographic operations.Manage key servers**

Note In an Enhanced Link Mode configuration, you must perform the backup on the vCenter Server that the key provider belongs to.

Procedure

1. Log in to the vCenter Server system with the vSphere Client.
2. Browse the inventory list and select the vCenter Server instance.
3. Click **Configure**, and under **Security** click **Key Providers**.
4. Select the vSphere Native Key Provider you want to back up.

   A status of "Not backed up" appears for key providers that you have not backed up.
5. Click **Back Up**.
6. To password-protect the backup, check the **Protect Native Key Provider data with password** box.
   a. Enter a password and save it in a secure location.
   b. Check the **I have saved the password in a secure place** box, indicating that you have saved the password to a secure place.
7. Click **Back Up Key Provider**.

   The backup file is in PKCS#12 format.
8. Save the backup file in a secure location.

Results

The status of the vSphere Native Key Provider changes from Not Backed Up, to Warning, to Active. Warning indicates that the vCenter Server is still pushing the information to all the ESXi hosts in the data center. Active means that the information has been pushed to all the hosts.

What to do next

To add vTPMs to your virtual machines, see Chapter 11 Securing Virtual Machines with Virtual Trusted Platform Module. To encrypt virtual machines, see Chapter 10 Use Encryption in Your vSphere Environment.

Recovering a vSphere Native Key Provider

You can recover the vSphere Native Key Provider either through the vSphere Client or from the vCenter Server Appliance Backup.

When necessary, you can recover a vSphere Native Key Provider in the following ways.

1 If you do not need to rebuild your vCenter Server Appliance, use the vSphere Client to restore the key provider. See Restore a vSphere Native Key Provider Using the vSphere Client.

2 If you must rebuild your vCenter Server Appliance, you must restore the key provider from your vCenter Server Appliance Backup. When you perform a vCenter Server Appliance Backup, it saves the Native Key Provider. See https://blogs.vmware.com/vsphere/2018/05/vcenter-server-appliance-6-7-file-based-backup-and-restore-walkthroughs.html for information about restoring the vCenter Server Appliance from backup.

Restore a vSphere Native Key Provider Using the vSphere Client

You can use the vSphere Client to restore the vSphere Native Key Provider.

You can restore a vSphere Native Key Provider in case it was accidentally deleted or if you must perform a disaster recovery.

When you restore a vSphere Native Key Provider, you do not need to back up the key provider again. The initial backup suffices. Continue to maintain the backup file in a secure location.

Note You can also use this task to configure vSphere Native Key Provider for vCenter Server systems in an Enhanced Linked Mode configuration. After you create the vSphere Native Key Provider on one vCenter Server system in the Enhanced Linked Mode configuration, use the **Restore** function to import the encrypted key file to the other ELM-connected vCenter Server systems.

Prerequisites

- Required privilege: **Cryptographic operations.Manage key servers**
- The key provider backup file.
- The password for the key provider file, if you entered one when you backed up the key provider.

Procedure

1 Log in to the vCenter Server system with the vSphere Client.

2 Browse the inventory list and select the vCenter Server instance.

3 Click **Configure**, and under **Security** click **Key Providers**.

4 Select the vSphere Native Key Provider and click **Restore**.

5 Browse to the file location and select the backup encrypted key file.

   The file was saved in PKCS#12 format.

6 (Optional) If the file is password protected, enter the password.

7 Click **Next**.

8 (Optional) If you decided to use this key provider only with TPM-protected ESXi hosts, select the check box.

9 Click **Finish**.

Results

The vSphere Native Key Provider is imported to the vCenter Server. To use the vSphere Native Key Provider for encryption tasks, ensure that you first select it in the **Key Provider** pane and click **Set as Default**.

Update a vSphere Native Key Provider

As part of your regular key rotation plans, you can use PowerCLI to update a vSphere Native Key Provider.

If you have a policy for key rotation, you can update the vSphere Native Key Provider and rekey the virtual machines that you encrypted with that key provider. You must use PowerCLI to update the vSphere Native Key Provider. You can also rekey the encrypted virtual machines without updating the key provider. In this case, only the virtual machine keys are changed. To rekey a virtual machine, see Rekey an Encrypted Virtual Machine Using the vSphere Client.

Prerequisites

- Required privilege: **Cryptographic operations.Manage key servers**
- PowerCLI 12.3.0

Procedure

1 In a PowerCLI session, run the `Connect-VIServer` cmdlet to connect as an administrator user to the vCenter Server where you configured the vSphere Native Key Provider that you want to update.

```
Connect-VIServer -server VC_ip_address -User admin_user -Password 'password'
```

2 To get your vSphere Native Key Provider names, run the `Get-KeyProvider` cmdlet with the optional `Type` parameter.

```
Get-KeyProvider -Type NativeKeyProvider
```

3 To update the key provider, run the `Set-KeyProvider` cmdlet, specifying your key provider name and GUID.

   You can generate a GUID to use by running the `New-Guid` cmdlet.

   ```
   Set-KeyProvider -KeyProvider KeyProvider_name -KeyId Guid
   ```

   A warning appears about backing up the configuration.

4 To back up the key provider, run the `Export-KeyProvider` cmdlet.

   ```
   Export-KeyProvider -KeyProvider KeyProvider_name -FilePath path_file_name
   ```

   You can also back up the key provider using the vSphere Client. See Back up a vSphere Native Key Provider.

Results

When a key provider is updated, its status changes to Not Backed Up. After you back up the key provider, its status changes to Active.

Delete a vSphere Native Key Provider

You can delete a vSphere Native Key Provider from vCenter Server.

After you delete a vSphere Native Key Provider, virtual machines that have vTPMs or that are encrypted continue to run. If you reboot the ESXi host, its encrypted virtual machines enter a locked state. After you unregister these virtual machines, they enter a locked state when you try to re-register them. The only way to unlock the virtual machines is to restore the previous vSphere Native Key Provider.

Prerequisites

Required privilege: **Cryptographic operations.Manage key servers**

Before you delete a vSphere Native Key Provider, rekey any encrypted virtual machines and datastores that were encrypted using that key provider to another key provider. See Rekey an Encrypted Virtual Machine Using the vSphere Client.

In addition, maintain a backup of the vSphere Native Key Provider in case you must rekey an encrypted virtual machine after deleting the key provider.

Procedure

1 Log in to the vCenter Server system with the vSphere Client.

2 Browse the inventory list and select the vCenter Server instance.

3 Click **Configure**, and under **Security** click **Key Providers**.

4 Select the key provider you want to delete.

5 Click **Delete**.

6 Read the warning message and slide the slider all the way to the right.

7 Click **Delete**.

Results

The vSphere Native Key Provider is removed from the vCenter Server.

vSphere Trust Authority

9

With vSphere 7.0 and later, you can take advantage of VMware® vSphere Trust Authority™. vSphere Trust Authority is a foundational technology that enhances workload security. vSphere Trust Authority establishes a greater level of trust in your organization by associating an ESXi host's hardware root of trust to the workload itself.

Read the following topics next:

- vSphere Trust Authority Concepts and Features
- Configuring vSphere Trust Authority
- Managing vSphere Trust Authority in Your vSphere Environment

vSphere Trust Authority Concepts and Features

vSphere Trust Authority secures your SDDC against malicious attacks by extending the trustworthiness of a trusted computing base to your organization's entire computing infrastructure. vSphere Trust Authority uses remote attestation and controlled access to advanced cryptographic capabilities.

vSphere Trust Authority is a set of services that satisfies high security requirements. With vSphere Trust Authority, you can set up and maintain a secure infrastructure. You can ensure that sensitive workloads run only on ESXi hosts proven to have booted authentic software.

How vSphere Trust Authority Protects Your Environment

You configure vSphere Trust Authority services to attest your ESXi hosts, which then become capable of performing trusted cryptographic operations.

vSphere Trust Authority uses remote attestation for ESXi hosts to prove the authenticity of their booted software. Attestation verifies that the ESXi hosts are running authentic VMware software, or VMware-signed partner software. Attestation relies on measurements that are rooted in a Trusted Platform Module (TPM) 2.0 chip installed in the ESXi host. In vSphere Trust Authority, an ESXi can access encryption keys and perform cryptographic operations only after it has been attested.

vSphere Trust Authority Glossary

vSphere Trust Authority introduces specific terms and definitions that are important to understand.

Table 9-1. vSphere Trust Authority Glossary

Term	Definition
VMware vSphere® Trust Authority™	Specifies a set of services that enables a Trusted Infrastructure. It is responsible for ensuring that ESXi hosts are running trusted software and for releasing encryption keys only to trusted ESXi hosts.
vSphere Trust Authority Components	The vSphere Trust Authority components are: ■ Attestation Service ■ Key Provider Service
Attestation Service	Attests the state of a remote ESXi host. Uses TPM 2.0 to establish a hardware root of trust, and verifies software measurements against a list of administrator approved ESXi versions.
Key Provider Service	Encapsulates one or more key servers and exposes Trusted Key Providers that can be specified when encrypting virtual machines. Currently, key servers are limited to the KMIP protocol.
Trusted Infrastructure	A Trusted Infrastructure consists of: ■ A Trust Authority vCenter Server ■ A Workload vCenter Server ■ At least one vSphere Trust Authority Cluster (configured as part of the Trust Authority vCenter Server) ■ At least one Trusted Cluster (configured as part of the Workload vCenter Server) ■ Encrypted workload virtual machines running in the Trusted Cluster ■ At least one KMIP-compliant key management server **Note** You must use separate vCenter Server systems for the Trust Authority Cluster and the Trusted Cluster.
Trust Authority Cluster	Consists of a vCenter Server cluster of ESXi hosts that run the vSphere Trust Authority components (the Attestation Service and the Key Provider Service).
Trust Authority Host	An ESXi host running vSphere Trust Authority components (the Attestation Service and the Key Provider Service).
Trusted Cluster	Consists of a vCenter Server cluster of Trusted ESXi hosts that are remotely attested by the Trust Authority Cluster. Although not strictly required, a configured Key Provider Service greatly increases the value provided by a Trusted Cluster.
Trusted Host	An ESXi host whose software has been validated by the Trust Authority Cluster Attestation Service. This host runs workload virtual machines that can be encrypted using Key Providers published by the Trust Authority Cluster Key Provider Service.
vSphere Encryption for Virtual Machines	With vSphere Virtual Machine Encryption, you can create encrypted virtual machines and encrypt existing virtual machines. ■ Starting in vSphere 6.5, vCenter Server requests keys from an external key server. The key server generates and stores the keys, and passes them to vCenter Server for distribution. ■ Starting in vSphere 7.0, the trusted connection can be set up between vSphere Trust Authority and a key server. This setup removes the need for the vCenter Server and workload ESXi hosts from requiring direct key server credentials, and enables an extra layer of defense-in-depth security.

Table 9-1. vSphere Trust Authority Glossary (continued)

Term	Definition
Trusted Key Provider	A Key Provider that encapsulates a single encryption key on a key server. Access to the encryption key requires the Attestation Service to acknowledge that the ESXi software has been verified on the Trusted Host.
Standard Key Provider	A Key Provider that gets encryption keys directly from a key server, and distributes keys to the required hosts in a data center. Previously referred to in vSphere as KMS Cluster.
Key Server	A KMIP key management server (KMS) that is associated with a Key Provider.
Workload vCenter Server	The vCenter Server that manages and is used to configure one or more Trusted Clusters.

vSphere Trust Authority Basics

With vSphere Trust Authority, you can:

- Provide ESXi hosts with a hardware root of trust and remote attestation capabilities
- Restrict encryption key management by releasing keys only to attested ESXi hosts
- Create a more secure administrative environment for managing trust
- Centralize management of multiple key servers
- Continue to perform cryptographic operations on virtual machines but with an enhanced level of encryption key management

In vSphere 6.5 and 6.7, virtual machine encryption depends on vCenter Server to obtain encryption keys from a key server and push them to ESXi hosts as needed. vCenter Server authenticates with the key server by using client and server certificates, which are stored in VMware Endpoint Certificate Store (VECS). Encryption keys that are sent from the key server pass through vCenter Server memory to the required ESXi hosts (with data encryption provided by TLS over the wire). In addition, vSphere depends on privilege checks in vCenter Server to validate user permissions and enforce key server access restrictions. Although this architecture is secure, it does not address the potential for a compromised vCenter Server, a rogue vCenter Server administrator, or a management or configuration error that might result in leaked or stolen secrets.

In vSphere 7.0, vSphere Trust Authority addresses these problems. You can create a trusted computing base, which consists of a secure, manageable set of ESXi hosts. vSphere Trust Authority implements a remote attestation service for the ESXi hosts you want to trust. Furthermore, vSphere Trust Authority improves upon TPM 2.0 attestation support (added to vSphere beginning in the 6.7 release), to implement access restrictions on encryption keys and so better protect virtual machine workload secrets. In addition, vSphere Trust Authority allows only authorized Trust Authority administrators to configure vSphere Trust Authority services, and configure Trust Authority hosts. The Trust Authority administrator can be the same user as the vSphere administrator user, or a separate user.

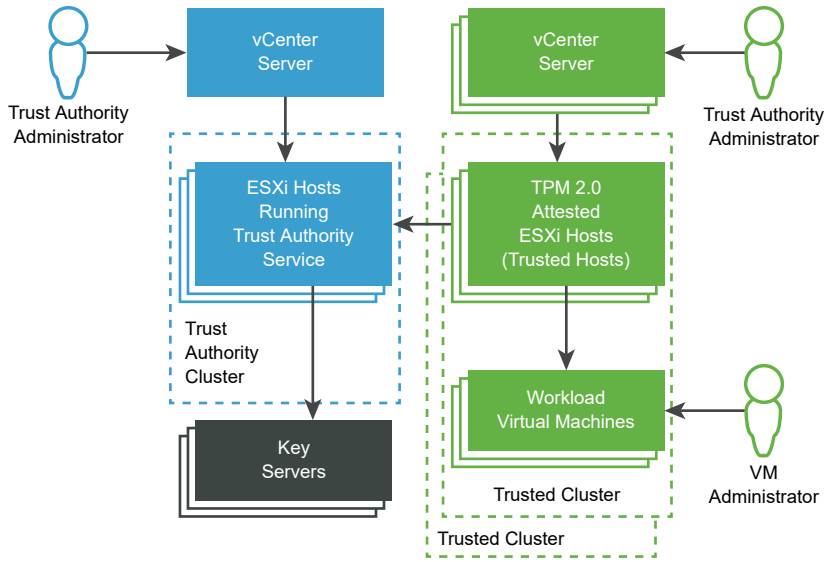
In the end, vSphere Trust Authority enables you to run your workloads in a more safe and secure environment by:

- Detecting tampering
- Disallowing unauthorized changes
- Preventing malware and modifications
- Limiting sensitive workloads to run only on a verified, safe hardware and software stack

vSphere Trust Authority Architecture

The following figure shows a simplified view of the vSphere Trust Authority architecture.

Figure 9-1. vSphere Trust Authority Architecture



In this figure:

1 vCenter Server systems

   Separate vCenter Server systems manage the Trust Authority Cluster and Trusted Clusters.

2 Trust Authority Cluster

   Consists of the ESXi hosts that run the vSphere Trust Authority components.

3 Key Servers

   Store encryption keys that are used by the Key Provider Service when encryption operations are performed. The key servers are external to vSphere Trust Authority.

4 Trusted Clusters

   Consist of the ESXi Trusted Hosts, which have been remotely attested with a TPM, and that run encrypted workloads.

5 Trust Authority Administrator

Administrator who is a member of the vCenter Server TrustedAdmins group, and configures the Trusted Infrastructure.

vSphere Trust Authority enables flexibility in how you designate Trust Authority administrators. The Trust Authority administrators in the figure can be separate users. It is also possible for the Trust Authority administrators to be the same user, using credentials that are linked across the vCenter Server systems. In this case, it is the same user and the same TrustedAdmins group.

6 VM Administrator

Administrator who has been granted privileges to manage the encrypted workload virtual machines on the Trusted Hosts.

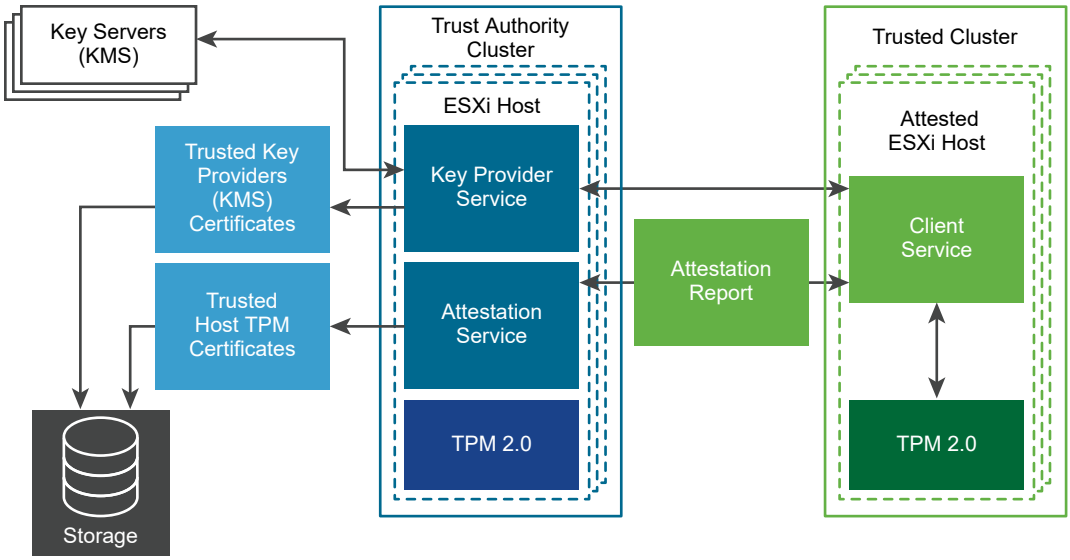
Trusted Infrastructure Overview

vSphere Trust Authority services, at least one external KMIP-compliant key server, the vCenter Server systems, and your ESXi hosts contribute to the Trusted Infrastructure.

What Is a Trusted Infrastructure

A Trusted Infrastructure consists of at least one vSphere Trust Authority Cluster, at least one Trusted Cluster, and at least one external KMIP-compliant key server. Each cluster contains ESXi hosts that run specific vSphere Trust Authority services, as shown in the following figure.

Figure 9-2. vSphere Trust Authority Services



Configuring the Trust Authority Cluster enables two services:

- Attestation Service
- Key Provider Service

When you configure vSphere Trust Authority, the ESXi hosts in the Trusted Cluster communicate with the Attestation Service. The Key Provider Service interposes between the Trusted Hosts and one or more trusted key providers.

Note Currently, the ESXi hosts in the Trust Authority Cluster do not require a TPM. However, as a matter of best practice, consider installing new ESXi hosts with TPMs.

About the vSphere Trust Authority Attestation Service

The Attestation Service generates a signed document that contains assertions describing the binary and configuration state of the remote ESXi hosts in the Trusted Cluster. The Attestation Service attests the state of the ESXi hosts using a Trusted Platform Module (TPM) 2.0 chip as its basis for software measurement and reporting. The TPM on the remote ESXi host measures the software stack and sends the configuration data to the Attestation Service. The Attestation Service verifies that the software measurement signature can be attributed to a previously configured trusted TPM endorsement key (EK). The Attestation Service also ensures that the software measurement matches one of a set of previously blessed ESXi images. The Attestation Service signs a JSON Web Token (JWT) that it issues to the ESXi host, providing the assertions about the identity, validity, and configuration of the ESXi host.

What Is the vSphere Trust Authority Key Provider Service

The Key Provider Service removes the need for the vCenter Server and the ESXi hosts from requiring direct key server credentials. In vSphere Trust Authority, for an ESXi host to have access to an encryption key, it must authenticate with the Key Provider Service.

For the Key Provider Service to connect to a key server, the Trust Authority administrator must configure a trust setup. For most KMIP-compliant servers, configuring a trust setup involves configuring client and server certificates.

To ensure that the keys are released only to ESXi Trusted Hosts, the Key Provider Service acts as a gatekeeper to the key servers. The Key Provider Service hides the key server specifics from the rest of the data center software stack by using the concept of a trusted key provider. Each trusted key provider has a single configured primary encryption key, and references one or more key servers. The Key Provider Service can have several configured trusted key providers. For example, you might want to have a separate trusted key provider for each department in an organization. Each trusted key provider uses a different primary key, but can reference the same backing key server.

After you create a trusted key provider, the Key Provider Service can accept requests from the ESXi Trusted Hosts to run cryptographic operations against that trusted key provider.

When an ESXi Trusted Host requests operations against a trusted key provider, the Key Provider Service makes sure that the ESXi host that is trying to obtain the encryption key is attested. After passing all the checks, the ESXi Trusted Host receives encryption keys from the Key Provider Service.

What Ports Are Used by vSphere Trust Authority

The vSphere Trust Authority services listen for connections behind the ESXi host's reverse proxy. All communication occurs over HTTPS on port 443.

What Are vSphere Trust Authority Trusted Hosts

The ESXi Trusted Hosts are configured to use trusted key providers to perform cryptographic operations. The ESXi Trusted Hosts perform key operations by communicating to the Key Provider Service and the Attestation Service. For authentication and authorization, the ESXi Trusted Hosts use a token obtained from the Attestation Service. To get a valid token, the ESXi Trusted Host must successfully attest to the Attestation Service. The token contains certain claims that are used to decide whether the ESXi Trusted Host is authorized to access a trusted key provider.

vSphere Trust Authority and Key Servers

vSphere Trust Authority requires the use of at least one key server. In previous vSphere releases, a key server was called a Key Management Server or KMS. Currently, vSphere virtual machine encryption solution supports KMIP 1.1 compliant key servers.

How Does vSphere Trust Authority Store Configuration and State Information

vCenter Server is mostly a pass-through service for vSphere Trust Authority configuration and state information. Most vSphere Trust Authority configuration and state information is stored on the ESXi hosts in the ConfigStore database. Some state information is stored in the vCenter Server database as well.

Note Because most vSphere Trust Authority configuration information is stored on the ESXi hosts, the vCenter Server file-based backup mechanism does not back up this information. To ensure the configuration information for your vSphere Trust Authority deployment is saved, see Backing Up the vSphere Trust Authority Configuration.

How Does vSphere Trust Authority Integrate with vCenter Server

You configure separate vCenter Server instances to manage the Trust Authority Cluster and Trusted Cluster. See Configuring vSphere Trust Authority.

On a Trusted Cluster, the vCenter Server manages the Trust Authority API calls and passes them through to the ESXi hosts. The vCenter Server replicates the API calls across all ESXi hosts in the Trusted Cluster.

After you configure vSphere Trust Authority initially, you can add or remove ESXi hosts to or from a Trust Authority Cluster or a Trusted Cluster. See Adding and Removing vSphere Trust Authority Hosts.

vSphere Trust Authority Process Flows

Understanding vSphere Trust Authority process flows is essential to learning how to configure and manage your Trusted Infrastructure.

How Do You Configure vSphere Trust Authority

vSphere Trust Authority is not activated by default. You must manually configure vSphere Trust Authority in your environment. See Configuring vSphere Trust Authority.

When you configure vSphere Trust Authority, you must specify which versions of ESXi software the Attestation Service accepts, and also which Trusted Platform Modules (TPMs) are trustworthy.

TPM and Attestation

This guide uses the following definitions when discussing TPMs and attestation.

Table 9-2. TPM and Attestation Glossary

Term	Definition
Endorsement Key (EK)	A TPM is manufactured with an RSA public/private key pair built into the hardware, called the endorsement key (EK). The EK is unique to a particular TPM.
EK Public Key	The public portion of the EK key pair.
EK Private Key	The private portion of the EK key pair.
EK Certificate	The EK public key wrapped with a signature. The EK certificate is created by the TPM manufacturer who uses its Certificate Authority private key to sign the EK public key. Not all TPMs contain an EK certificate. In this case, the EK public key is not signed.
TPM Attestation	The ability of the Attestation Service to verify the software that is running on a remote host. TPM attestation is accomplished through cryptographic measurements made by the TPM while the remote host starts, and is relayed to the Attestation Service on request. The Attestation Service establishes trust in the TPM through either the EK public key or the EK certificate.

Configuring TPM Trust on the Trusted Hosts

An ESXi Trusted Host must contain a TPM. A TPM is manufactured with a public/private key pair built into the hardware, called the endorsement key (EK). Although TPM 2.0 permits many key/certificate pairs, the most common is an RSA-2048 key pair. When a TPM EK public key is signed by a CA, the result is the EK certificate. The TPM manufacturer typically pre-generates at least one EK, signs the public key with a Certificate Authority, and embeds the signed certificate in the TPM's non-volatile memory.

You can configure the Attestation Service to trust TPMs as follows:

- Trust all CA certificates with which the manufacturer signed the TPM (the EK public key). The default setting for the Attestation Service is to trust CA certificates. In this approach, the same CA certificate covers many ESXi hosts, and so reduces your administrative overhead.

- Trust the ESXi host's TPM CA certificate and EK public key. The latter can be either the EK certificate or the EK public key. Although this approach provides more security, it requires you to configure information about each Trusted Host.
- Some TPMs do not contain an EK certificate. In this case, you trust the EK public key.

Deciding to trust all TPM CA certificates is operationally convenient. You configure new certificates only when you add a new class of hardware to your data center. By trusting individual EK certificates, you can limit access to specific ESXi hosts.

You can also decide not to trust TPM CA certificates. Though an uncommon situation, you can use this configuration when an EK is not signed by a CA. Currently, this functionality is not fully implemented.

Note Some TPMs do not include EK certificates. If you want to trust individual ESXi hosts, the TPM must include an EK certificate.

Attesting TPMs

To begin the attestation process, the ESXi Trusted Host in the Trusted Cluster sends the pre-configured EK public key and EK certificate to the Attestation Service on the Trust Authority Cluster. When the Attestation Service receives the request, it looks up the EK in its configuration, which can be the EK public key or the EK certificate, or both, depending on the configuration. If no case is valid, the Attestation Service rejects the attestation request.

The EK is not directly used for signing, so an Attestation Key (AK or AIK) is negotiated. The negotiation protocol ensures that a newly created AK is bound to the previously verified EK, preventing a man-in-the-middle situation, or an impersonator. After an AK is negotiated, it is reused on future attestation requests rather than generating a new one each time.

The ESXi Trusted Host reads the Quote and PCR values from the TPM. The Quote is signed by the AK. The ESXi Trusted Host also reads the TCG Event Log, which includes all the events that resulted in the current PCR state. This TPM information is sent to the Attestation Service for validation. The Attestation Service verifies the PCR values using the event log.

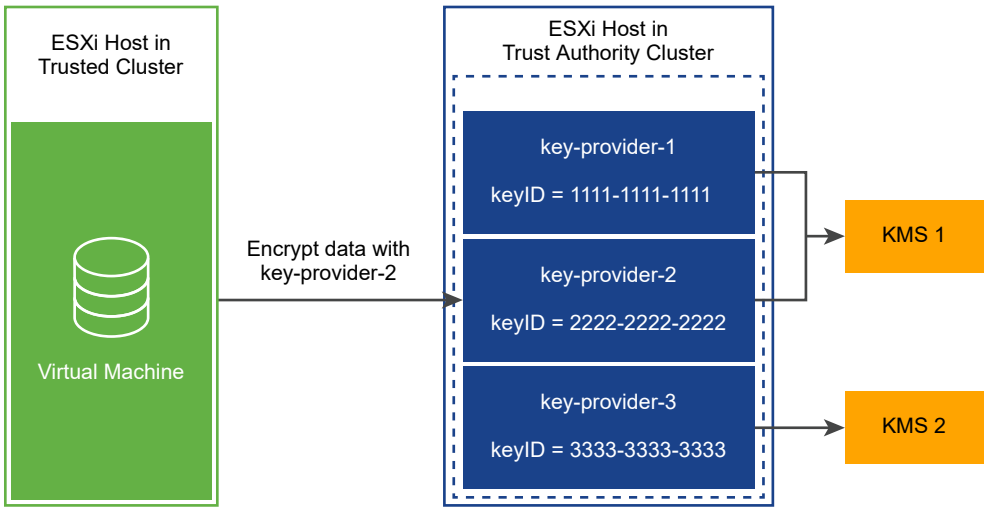
How Do Key Providers Work with Key Servers

The Key Provider Service uses the concept of a trusted key provider to hide the key server specifics from the rest of the data center software. Each trusted key provider has a single configured primary encryption key, and references one or more key servers. The primary encryption key is present in the key servers. As part of configuring vSphere Trust Authority, you must provision the primary key as a separate activity and activate it. The Key Provider Service can have several configured trusted key providers. Each trusted key provider uses a different primary key, but can reference the same backing key server.

When a new trusted key provider is added, the Trust Authority administrator must specify the key server and an existing key identifier on that key server.

The following figure shows the relationship between the Key Provider Service and key servers.

Figure 9-3. Key Provider and Key Server



After you configure a trusted key provider for a Trusted Cluster, the Key Provider Service can accept requests to run cryptographic operations against that trusted key provider. For example, in this figure, three trusted key providers are configured, two for KMS-1, and one for KMS-2. The Trusted Host requests an encryption operation against key-provider-2. The Trusted Host requests an encryption key to be generated and returned, and uses this encryption key to perform encryption operations.

The Key Provider Service uses the primary key referenced by key-provider-2 to encrypt the specified plaintext data and return the corresponding ciphertext. Later, the Trusted Host can provide the same ciphertext to a decrypt operation and get back the original plaintext.

vSphere Trust Authority Authentication and Authorization

vSphere Trust Authority administrative operations require a user that is a member of the TrustedAdmins group. Having only Trust Authority administrator privileges is not sufficient to perform all administrative operations that involve the ESXi hosts. For more information, see Prerequisites and Required Privileges for vSphere Trust Authority.

Adding a Trusted Host to a Trusted Cluster

The steps to add ESXi hosts initially to the Trusted Cluster are described in Configuring vSphere Trust Authority.

Later, if you want to add ESXi hosts to the Trusted Cluster, the workflow is different. See Adding and Removing vSphere Trust Authority Hosts.

When you initially add ESXi hosts to the Trusted Cluster, you must gather the following information:

- TPM certificate for each type of hardware in the cluster
- ESXi image for each version of ESXi in the cluster
- vCenter Server principal information

If you later add ESXi hosts to a Trusted Cluster, you might need to collect some additional information. That is, if the new ESXi hosts differ in hardware or ESXi version from the original hosts, you must collect the new ESXi host information and import it to the Trust Authority Cluster. You only must collect the vCenter Server principal information one time per vCenter Server system.

vSphere Trust Authority Topology

vSphere Trust Authority requires separate vCenter Server systems for the Trust Authority Cluster and the Trusted Cluster.

The Trust Authority Cluster is configured and managed on an independent, isolated vCenter Server. The vCenter Server of the Trust Authority Cluster cannot also be the vCenter Server of the Trusted Cluster. The Trusted Cluster must have its own, separate vCenter Server. A single vCenter Server can manage multiple Trusted Clusters. Multiple vCenter Server systems for Trusted Clusters can participate in enhanced linked mode. The vCenter Server for the Trust Authority Cluster cannot participate in enhanced linked mode with other Trust Authority Cluster vCenter Server systems or Trusted Cluster vCenter Server systems.

The Trust Authority administrator manages the Trust Authority Cluster and its associated vCenter Server independently from other vCenter Server instances, because this approach provides the best security isolation.

The Trust Authority administrator documents or publishes the hostnames and SSL certificates that Trusted Cluster administrators use to configure their clusters. The Trust Authority administrator also provisions trusted key providers for the organization and its departments, or even individual administrators.

You cannot deploy vSphere Trust Authority services directly on the Trusted Cluster managed by the Workload vCenter Server, because the workload administrator has high privilege access to the ESXi hosts. This type of deployment does not achieve the necessary separation of roles that is required to meet the security objectives of vSphere Trust Authority.

Prerequisites and Required Privileges for vSphere Trust Authority

You must consider hardware and software requirements when configuring vSphere Trust Authority. You must set cryptographic privileges and roles to use encryption. The user who performs vSphere Trust Authority tasks must have the appropriate privileges.

Requirements for vSphere Trust Authority

To use vSphere Trust Authority, your vSphere environment must meet these requirements:

- ESXi Trusted Host hardware requirements:
  - TPM 2.0
  - Secure boot must be enabled

  - EFI firmware
- Component requirements:
  - vCenter Server 7.0 or later
  - A dedicated vCenter Server system for the vSphere Trust Authority Cluster and ESXi hosts
  - A separate vCenter Server system for the Trusted Cluster and ESXi Trusted Hosts
  - A key server (called a Key Management Server, or KMS, in prior vSphere releases)
- Virtual machine requirements:
  - EFI firmware
  - Secure Boot Enabled

Note Before you can begin configuring vSphere Trust Authority, ensure that you have set up your vCenter Server systems for the Trust Authority Cluster and Trusted Cluster, and added ESXi hosts to each cluster.

Cryptography Privileges

vSphere Trust Authority does not introduce any new cryptography privileges. The same cryptography privileges described in Cryptography Privileges and Roles apply to vSphere Trust Authority.

Host Encryption Mode

vSphere Trust Authority does not introduce any new requirements for enabling host encryption mode on the ESXi Trusted Hosts. See Prerequisites and Required Privileges for Encryption Tasks for more information about host encryption mode.

About the vSphere Trust Authority Roles and the TrustedAdmins Group

vSphere Trust Authority operations require a user that is a member of the TrustedAdmins group. This user is called the Trust Authority administrator. vSphere administrators must either add themselves to the TrustedAdmins group or add other users to the group to gain the Trusted Infrastructure administrator role. The Trusted Infrastructure administrator role is necessary for vCenter Server authorization. The TrustedAdmins group is necessary for authentication on the ESXi hosts that are part of the Trusted Infrastructure. Users with the **Cryptographic Operations.Register host** privilege on ESXi hosts can manage the Trusted Cluster. The vCenter Server permissions are not propagated to the Trust Authority hosts, only to the Trusted Hosts. Only members of the TrustedAdmins group are granted privileges on the Trust Authority hosts. Group membership is verified on the ESXi host itself.

Note vSphere administrators and members of the Administrators group are assigned the Trusted Infrastructure administrator role, but this role by itself does not permit a user to perform vSphere Trust Authority operations. Membership in the TrustedAdmins group is also required.

After vSphere Trust Authority is enabled, Trust Authority administrators can assign trusted key providers to Trusted Hosts. Those Trusted Hosts can then use the trusted key providers to perform cryptographic tasks.

In addition to the Trusted Infrastructure administrator role, vSphere Trust Authority provides the No Trusted Infrastructure administrator role, which contains all privileges in vCenter Server except the ones that call the vSphere Trust Authority APIs.

vSphere Trust Authority groups, roles, and users function as follows:

- On first boot, vSphere grants the TrustedAdmins group the Trusted Infrastructure administrator role, which has global permissions.
- The Trusted Infrastructure administrator role is a system role that has the required privileges to call the vSphere Trust Authority APIs (`TrustedAdmin.*`), and the system privileges **System.Read**, **System.View**, and **System.Anonymous** to view inventory objects.
- The No Trusted Infrastructure administrator role is a system role that contains all privileges in vCenter Server except the ones to call the vSphere Trust Authority APIs. Adding new privileges to vCenter Server also adds them to the No Trusted Infrastructure administrator role. (The No Trusted Infrastructure administrator role is similar to the No cryptography administrator role.)
- The vSphere Trust Authority privileges (`TrustedAdmin.*` APIs) are not included in the No cryptography administrator role, preventing users with this role from setting up a Trusted Infrastructure or performing cryptographic operations.

The use cases for these users, groups, and roles, are shown in the following table.

Table 9-3. vSphere Trust Authority Users, Groups, and Roles

User, Group, or Role	Can Call vSphere Trust Authority vCenter Server API (Includes Calls to vSphere Trust Authority ESXi API)	Can Call vSphere Trust Authority vCenter Server API (Does Not Include Calls to vSphere Trust Authority ESXi API)	Can Perform Host Operations in Cluster Not Related to vSphere Trust Authority	Comment
User in both Administrators@*system.domain* group and TrustedAdmins@*system.domain* group	Yes	Yes	Yes	NA
User in TrustedAdmins@*system.domain* group only	Yes	Yes	No	Such a user cannot perform regular cluster management operations.
User in Administrators@*system.domain* group only	Yes	No	Yes	NA

Table 9-3. vSphere Trust Authority Users, Groups, and Roles (continued)

User, Group, or Role	Can Call vSphere Trust Authority vCenter Server API (Includes Calls to vSphere Trust Authority ESXi API)	Can Call vSphere Trust Authority vCenter Server API (Does Not Include Calls to vSphere Trust Authority ESXi API)	Can Perform Host Operations in Cluster Not Related to vSphere Trust Authority	Comment
User with Trusted Infrastructure administrator role but not in TrustedAdmins@*system.domain* group	Yes	No	No	The ESXi host checks the group membership of the user to grant permissions.
User with No Trusted Infrastructure administrator role only	No	No	Yes	Such a user is similar to an administrator who cannot perform vSphere Trust Authority operations.

vSphere Trust Authority Best Practices, Caveats, and Interoperability

The vSphere Trust Authority architecture results in some additional recommendations. As you are planning your vSphere Trust Authority strategy, consider interoperability limitations.

Trusted Infrastructure Interoperability

For ESXi versions, the Attestation Service is backward and forward compatible. For example, you can have a cluster of ESXi hosts running ESXi 7.0 in the vSphere Trust Authority Cluster, and upgrade or patch ESXi hosts in the Trusted Cluster to a newer ESXi version. Similarly, you can upgrade or patch the ESXi hosts in the Trust Authority Cluster while keeping the ESXi hosts in the Trusted Cluster at the current version.

You cannot have a cluster function as both a Trust Authority Cluster and a Trusted Cluster. This configuration is not supported.

Trusted Cluster Configuration Limitation

You can configure only one Trusted Cluster per workload vCenter Server. A Trusted Cluster cannot be configured to reference multiple Trust Authority Clusters.

Supported Features

vSphere Trust Authority supports the following:

- vCenter High Availability (vCenter HA)
- VMware vSphere High Availability
- DRS
- DPM

- SRM, with the following understanding:
  - SRM with array-based replication is supported, if the same vSphere Trust Authority services configuration is available on the recovery side.
  - SPPG
- VADP
  - Support is the same as with standard encryption. Hot-add and NFC modes are supported, but not SAN mode. Backups are decrypted. VADP partners have the option of recovering the backed-up virtual machine with the same encryption key as the original virtual machine.
- vSAN
  - Virtual machine encryption is fully supported on top of vSAN.
- OVF
  - Encrypted virtual machines cannot be exported to OVF. However, virtual machines can be encrypted while being imported from an OVF.
- VVol

Unsupported Features

Currently, vSphere Trust Authority does not support the following:

- vSAN encryption
- First Class Disk (FCD) encryption
- vSphere Replication
- vSphere Host Profiles

vSphere Trust Authority Life Cycle

The vSphere Trust Authority services are packaged and installed as part of the base ESXi image.

Starting and Stopping Services

In the vSphere Client, you can start, stop, and restart vSphere Trust Authority services that are running on an ESXi host. You can restart services upon a configuration change or if you suspect functional or performance problems. To restart the service on an ESXi Trusted Host, you must log in to the host itself to restart the service. See Start, Stop, and Restart vSphere Trust Authority Services.

Upgrading and Patching

Each time you upgrade or patch an ESXi Trusted Host, you must update the vSphere Trust Authority Cluster with the new ESXi version information. One way to do so is to upgrade or patch a test ESXi host, export the ESXi base image information, import the image file to the Trust Authority Cluster, then upgrade or patch the ESXi Trusted Hosts.

Upgrading Best Practices

Best practice for upgrading a vSphere Trust Authority infrastructure is to upgrade the Trust Authority vCenter Server and Trust Authority Hosts first. In this way, you get the most benefit from the latest vSphere Trust Authority features. However, you can perform separate, standalone upgrades of vCenter Server and ESXi hosts to fit specific business reasons.

In general, follow this order for upgrading your vSphere Trust Authority infrastructure:

1 Upgrade the Trust Authority Cluster vCenter Server.

2 Upgrade the Trust Authority Hosts.

3 Upgrade the Trusted Cluster vCenter Server.

4 Upgrade the Trusted Hosts.

To ensure a smooth process, upgrade your Trust Authority Hosts and Trusted Hosts gradually, one-by-one.

Troubleshooting Upgrade Problems

If you encounter an unsuccessful upgrade of a Trust Authority Host, follow these steps.

1 Remove the Trust Authority Host from the Trusted Cluster.

2 Revert to the previous version of ESXi.

3 Re-add the Trust Authority Host to the cluster as described in the VMware knowledge base article at https://kb.vmware.com/s/article/77234.

4 Verify that the Trust Authority Host's configuration is consistent with the other Trust Authority Hosts in the Trust Authority Cluster. See Check Trusted Cluster Health.

When you upgrade to a new version of ESXi on a Trusted Host, attestation fails until you update the Trust Authority Cluster with the new ESXi base image information. This behavior is to be expected. You can no longer encrypt virtual machines or use existing virtual machines that were encrypted before upgrade until you fix the problem. Attestation error messages appear in the vSphere Client **Recent Tasks** pane and the `attestd.log`, `kmxa.log`, and `vpxd.log` files.

To correct the problem, follow these steps.

1 Run the `Export-VMHostImageDb` cmdlet to re-export the ESXi base images. See Step 5 in Collect Information About ESXi Hosts and vCenter Server to Be Trusted.

2 Run the `New-TrustAuthorityVMHostBaseImage` cmdlet to reimport the new base image to the vCenter Server of the Trust Authority Cluster. See Step 8 in Import the Trusted Host Information to the Trust Authority Cluster.

3 If you no longer must attest the older versions of ESXi (all the Trusted Hosts have been upgraded), run the `Remove-TrustAuthorityVMHostBaseImage` cmdlet to remove the versions. For example:

```
$vTA = Get-TrustAuthorityCluster 'vTA Cluster'
$baseImages = Get-TrustAuthorityVMHostBaseImage -TrustAuthorityCluster $vTA
Remove-TrustAuthorityVMHostBaseImage -VMHostBaseImage $baseImages
```

Backing Up the vSphere Trust Authority Configuration

Because most vSphere Trust Authority configuration information is stored on the ESXi hosts, the vCenter Server Backup does not back up this vSphere Trust Authority information. See Backing Up the vSphere Trust Authority Configuration.

Configuring vSphere Trust Authority

vSphere Trust Authority is not enabled by default. You must configure your environment for vSphere Trust Authority before you can start using it.

You enable vSphere Trust Authority services on a dedicated vCenter Server cluster, known as the vSphere Trust Authority Cluster. The Trust Authority Cluster acts as a centralized, secure management platform. You then enable a workload vCenter Server cluster as the Trusted Cluster. The Trusted Cluster contains the ESXi Trusted Hosts.

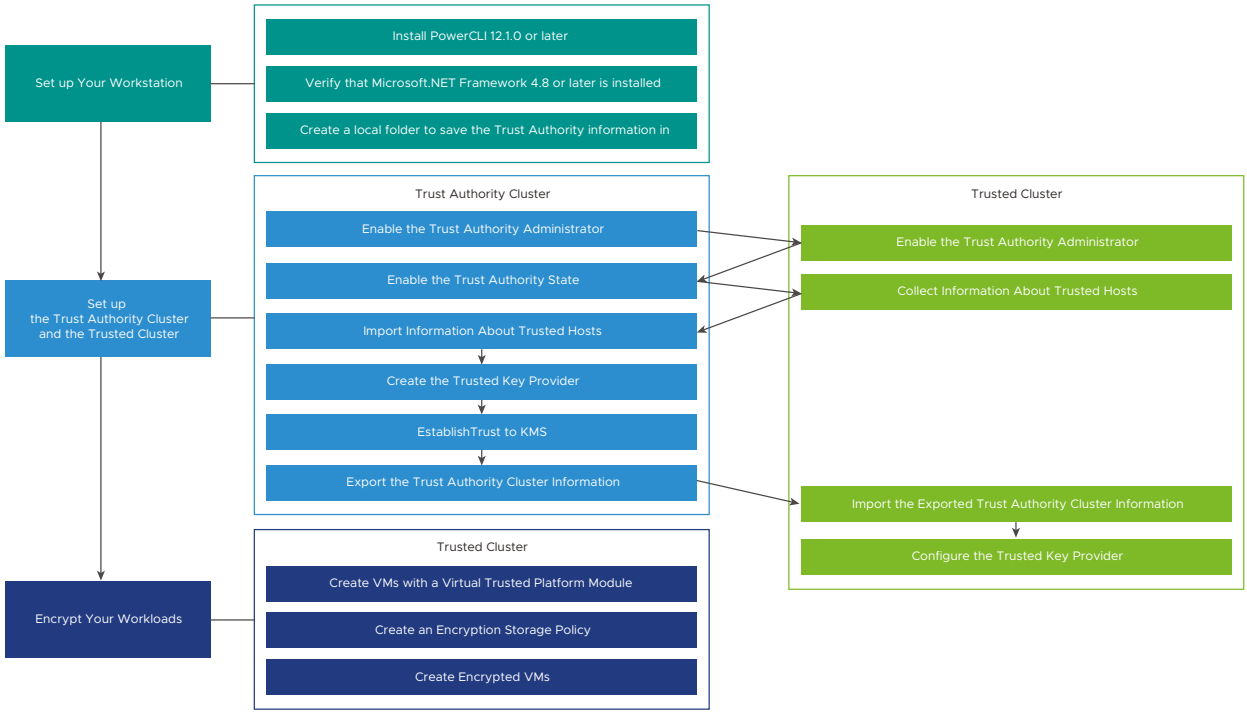
The Trust Authority Cluster attests the ESXi hosts in the Trusted Cluster remotely. The Trust Authority Cluster releases encryption keys only to attested ESXi hosts in the Trusted Cluster to encrypt virtual machines and virtual disks using trusted key providers.

Before you begin configuring vSphere Trust Authority, see Prerequisites and Required Privileges for vSphere Trust Authority for information on the required setup of vCenter Server systems and ESXi hosts.

You manage different aspects of vSphere Trust Authority in the following ways.

- Configure the vSphere Trust Authority services and trusted connections using PowerCLI cmdlets or the vSphere APIs. See *VMware PowerCLI Cmdlets Reference* and *vSphere Automation SDKs Programming Guide*.
- Manage the configuration of trusted key providers using the PowerCLI cmdlets or from the vSphere Client.
- Perform encryption workflows, as in prior vSphere releases, using the vSphere Client and APIs.

Figure 9-4. vSphere Trust Authority Workflow



To configure and manage vSphere Trust Authority, you use VMware PowerCLI, though some functionality is available in the vSphere Client.

When you configure vSphere Trust Authority, you must complete setup tasks on both the Trust Authority Cluster and the Trusted Cluster. Some of these tasks are order-specific. Use the task sequence outlined in this guide.

Note When adding more ESXi hosts to the Trusted Cluster after completing the initial vSphere Trust Authority setup, you might need to export and import the Trusted Host information again. That is, if the new ESXi hosts differ from the original hosts, you must collect the new ESXi host information and import it to the Trust Authority Cluster. See Adding and Removing vSphere Trust Authority Hosts.

What to read next

Procedure

1 Set up Your Workstation

To configure a vSphere Trust Authority deployment, you must first prepare a workstation with the necessary software and setup.

2 Enable the Trust Authority Administrator

To enable vSphere Trust Authority, you must add a user to the vSphere TrustedAdmins group. This user becomes the Trust Authority administrator. You use the Trust Authority administrator for most vSphere Trust Authority configuration tasks.

3 Enable the Trust Authority State

Making a vCenter Server cluster into a vSphere Trust Authority Cluster (also called enabling the Trust Authority State) starts the required Trust Authority services on the ESXi hosts in the cluster.

4 Collect Information About ESXi Hosts and vCenter Server to Be Trusted

To establish trust, the vSphere Trust Authority Cluster requires information about the Trusted Cluster's ESXi hosts and vCenter Server. You export this information as files for importing into the Trust Authority Cluster. You must ensure to keep these files confidential and transport them securely.

5 Import the Trusted Host Information to the Trust Authority Cluster

You import the exported ESXi host and vCenter Server information into the vSphere Trust Authority Cluster, so that the Trust Authority Cluster knows which hosts it can attest.

6 Create the Key Provider on the Trust Authority Cluster

For the Key Provider Service to connect to a key provider, you must create a trusted key provider then configure a trust setup between the vSphere Trust Authority Cluster and the key server (KMS). For most KMIP-compliant key servers, this configuration involves setting up client and server certificates.

7 Export the Trust Authority Cluster Information

For the Trusted Cluster to connect to the vSphere Trust Authority Cluster, you must export the Trust Authority Cluster's service information in the form of a file then import that file to the Trusted Cluster. You must ensure to keep this file confidential and transport it securely.

8 Import the Trust Authority Cluster Information to the Trusted Hosts

After you have imported the vSphere Trust Authority Cluster information to the Trusted Cluster, the Trusted Hosts start the attestation process with the Trust Authority Cluster.

9 Configure the Trusted Key Provider for Trusted Hosts Using the vSphere Client

You can configure the trusted key provider by using the vSphere Client.

10 Configure the Trusted Key Provider for Trusted Hosts Using the Command Line

You can configure trusted key providers by using the command line. You can configure the default trusted key provider for the vCenter Server, or at the cluster or the folder level in the vCenter object hierarchy.

Set up Your Workstation

To configure a vSphere Trust Authority deployment, you must first prepare a workstation with the necessary software and setup.

Perform the following steps on a workstation that has access to your vSphere Trust Authority environment.

Procedure

1 Install PowerCLI 12.1.0 or later. See *PowerCLI User's Guide*.

2 Verify that Microsoft .NET Framework 4.8 or later is installed.

3 Create a local folder in which to save the Trust Authority information that you export as files.

What to do next

Continue with Enable the Trust Authority Administrator.

Enable the Trust Authority Administrator

To enable vSphere Trust Authority, you must add a user to the vSphere TrustedAdmins group. This user becomes the Trust Authority administrator. You use the Trust Authority administrator for most vSphere Trust Authority configuration tasks.

Use a separate user from the vCenter Server administrator as your Trust Authority administrator. Having a separate user enhances the security of your environment. You must enable a Trust Authority administrator for both the Trust Authority Cluster and the Trusted Cluster.

Prerequisites

Either create or a user, or identify an existing user, to be the Trust Authority administrator.

Procedure

1 Connect to the vCenter Server of the Trust Authority Cluster by using the vSphere Client.

2 Log in as an administrator.

3 From the **Home** menu, select **Administration**.

4 Under **Single Sign On**, click **Users and Groups**.

5 Click **Groups** and click the **TrustedAdmins** group.

If the TrustedAdmins group does not appear initially, use the **Filter** icon to filter for it, or navigate through the groups by clicking the right arrow at the bottom of the pane.

6 In the **Group Members** area, click **Add Members**.

Make sure that the local identity source is selected (vsphere.local is the default, but you might have selected a different domain during installation), and search for the member (user) you want to add to the group as the Trust Authority administrator.

7 Select the member.

8 Click **Save**.

9 Repeat steps 1 through 8 for the vCenter Server of the Trusted Cluster.

What to do next

Continue with Enable the Trust Authority State.

Enable the Trust Authority State

Making a vCenter Server cluster into a vSphere Trust Authority Cluster (also called enabling the Trust Authority State) starts the required Trust Authority services on the ESXi hosts in the cluster.

Prerequisites

- Enable the Trust Authority Administrator.

Procedure

1. In a PowerCLI session, run the `Connect-VIServer` cmdlet to connect as the Trust Authority administrator user to the vCenter Server of the Trust Authority Cluster.

   ```
   Connect-VIServer -server TrustAuthorityCluster_VC_ip_address -User trust_admin_user
   -Password 'password'
   ```

2. To check the current state of the cluster, run the `Get-TrustAuthorityCluster` cmdlet.

   For example, this command shows the cluster, `vTA Cluster`, and that its state is disabled.

   ```
   Get-TrustAuthorityCluster

   Name                 State                Id
   ----                 -----                --
   vTA Cluster          Disabled             TrustAuthorityCluster-domain-c8
   ```

   The output shows either Disabled or Enabled in the State column for each cluster found. Disabled means that the Trust Authority services are not running.

3. To enable the Trust Authority Cluster, run the `Set-TrustAuthorityCluster` cmdlet.

   For example, this command enables the cluster `vTA Cluster`.

   ```
   Set-TrustAuthorityCluster -TrustAuthorityCluster 'vTA Cluster' -State Enabled
   ```

   The system responds with a confirmation prompt.

   ```
   Confirmation
   Setting TrustAuthorityCluster 'vTA Cluster' with new State 'Enabled'. Do you want to
   proceed?
   [Y] Yes  [A] Yes to All  [N] No  [L] No to All  [S] Suspend  [?] Help (default is "Y"):
   ```

4. At the confirmation prompt, press Enter. (The default is **Y**.)

   The output shows the state of the cluster. For example, the following shows that cluster `vTA Cluster` has been enabled:

   ```
   Name                 State                Id
   ----                 -----                --
   vTA Cluster          Enabled              TrustAuthorityCluster-domain-c8
   ```

Results

Two services start on the ESXi hosts in the Trust Authority Cluster: the Attestation Service and the Key Provider Service.

Example: Enable the Trusted State on the Trust Authority Cluster

This example shows how to use PowerCLI to enable services on the Trust Authority Cluster. The following table shows the example components and values that are used.

Table 9-4. Example vSphere Trust Authority Setup

Component	Value
vCenter Server for Trust Authority Cluster	192.168.210.22
Trust Authority Cluster name	vTA Cluster
Trust Authority administrator	trustedadmin@vsphere.local

```
PS C:\Users\Administrator.CORP> Disconnect-VIServer -server * -Confirm:$false
PS C:\Users\Administrator.CORP> Connect-VIServer -server 192.168.210.22 -User
trustedadmin@vsphere.local -Password 'VMware1!'

Name                           Port  User
----                           ----  ----
192.168.210.22                 443   VSPHERE.LOCAL\trustedadmin

PS C:\Users\Administrator.CORP> Get-TrustAuthorityCluster

Name                 State               Id
----                 -----               --
vTA Cluster          Disabled            TrustAuthorityCluster-domain-c8

PS C:\Users\Administrator.CORP> Set-TrustAuthorityCluster -TrustAuthorityCluster 'vTA
Cluster' -State Enabled

Confirmation
Setting TrustAuthorityCluster 'vTA Cluster' with new State 'Enabled'. Do you want to proceed?
[Y] Yes  [A] Yes to All  [N] No  [L] No to All  [S] Suspend  [?] Help (default is "Y"): y

Name                 State               Id
----                 -----               --
vTA Cluster          Enabled             TrustAuthorityCluster-domain-c8
```

What to do next

Continue with Collect Information About ESXi Hosts and vCenter Server to Be Trusted.

Collect Information About ESXi Hosts and vCenter Server to Be Trusted

To establish trust, the vSphere Trust Authority Cluster requires information about the Trusted Cluster's ESXi hosts and vCenter Server. You export this information as files for importing into

the Trust Authority Cluster. You must ensure to keep these files confidential and transport them securely.

You use vSphere Trust Authority PowerCLI cmdlets to export the following information as files from the ESXi hosts in the Trusted Cluster for the Trust Authority Cluster to know what software and hardware to trust.

- ESXi version
- TPM manufacturer (CA certificate)
- (Optional) Individual TPM (EK certificate)

Note Store these exported files in a secure location, in case you must restore the vSphere Trust Authority configuration.

If you have hosts of the same type and vendor, and manufactured during the same timeframe and location, you might be able to trust all TPMs by obtaining the CA certificate of only one of the TPMs. To trust an individual TPM, you obtain the EK certificate of the TPM.

You must also obtain the principal information from the Trusted Cluster's vCenter Server. The principal information contains the vpxd solution user and its certificate chain. The principal information enables the Trusted Cluster's vCenter Server to discover the available trusted key providers configured on the Trust Authority Cluster.

To configure vSphere Trust Authority initially, you must collect the ESXi version and TPM information. Also, you must collect the ESXi version each time after you deploy a new version of ESXi, including when you upgrade or apply a patch.

You collect the vCenter Server principal information only one time per vCenter Server system.

Prerequisites

- Identify the ESXi versions and TPM hardware types that are in the Trusted Cluster, and whether you want to trust all TPM hardware types, only certain ones, or individual hosts.
- On the machine from which you run the PowerCLI cmdlets, create a local folder in which to save the information you export as files.
- Enable the Trust Authority Administrator.
- Enable the Trust Authority State.

Procedure

1 In a PowerCLI session, run the following commands to disconnect any current connection and connect as the root user to one of the ESXi hosts in the Trusted Cluster.

```
Disconnect-VIServer -server * -Confirm:$false
Connect-VIServer -server host_ip_address -User root -Password 'password'
```

2 Run the `Get-VMHost` cmdlet to confirm the ESXi host.

```
Get-VMHost
```

The host information is displayed.

3 Assign `Get-VMHost` to a variable.

For example:

```
$vmhost = Get-VMHost
```

4 Run the `Export-Tpm2CACertificate` cmdlet to export the CA certificate of a given TPM manufacturer.

a Assign `Get-Tpm2EndorsementKey -VMHost $vmhost` to a variable.

For example, this command assigns `Get-Tpm2EndorsementKey -VMHost $vmhost` to the variable `$tpm2`.

```
$tpm2 = Get-Tpm2EndorsementKey -VMHost $vmhost
```

b Run the `Export-Tpm2CACertificate` cmdlet.

For example, this command exports the TPM certificate to the `cacert.zip` file. Ensure that the destination directory exists before running this command.

```
Export-Tpm2CACertificate -Tpm2EndorsementKey $tpm2 -FilePath C:\vta\cacert.zip
```

The file is created.

c Repeat for each TPM hardware type in the cluster that you want to trust. Use a different file name for each TMP hardware type so that you do not overwrite a previously exported file.

5 Run the `Export-VMHostImageDb` cmdlet to export the ESXi host description of software (the ESXi image).

For example, this command exports the information to the `image.tgz` file. Ensure that the destination directory exists before running this command.

```
Export-VMHostImageDb -VMHost $vmhost -FilePath C:\vta\image.tgz
```

Note The `Export-VMHostImageDb` cmdlet also works if you prefer to log in to the vCenter Server of the Trusted Cluster.

The file is created.

Repeat for each ESXi version in the cluster that you want to trust. Use a different file name for each version so that you do not overwrite a previously exported file.

6 Export the Trusted Cluster's vCenter Server principal information.

a Disconnect from the ESXi host.

```
Disconnect-VIServer -server * -Confirm:$false
```

b Connect to the vCenter Server of the Trusted Cluster using the Trust Authority administrator user. (Alternatively, you can use a user that has **Administrator** privileges.)

```
Connect-VIServer -server TrustedCluster_VC_ip_address -User trust_admin_user -Password
'password'
```

c To export the Trusted Cluster's vCenter Server principal information, run the `Export-TrustedPrincipal` cmdlet.

For example, this command exports the information to the `principal.json` file. Ensure that the destination directory exists before running this command.

```
Export-TrustedPrincipal -FilePath C:\vta\principal.json
```

The file is created.

7 (Optional) If you want to trust an individual host, you must export the TPM EK public key certificate.

See Export and Import a TPM Endorsement Key Certificate.

Results

The following files are created:

- TPM CA certificate file (.zip file extension)
- ESXi image file (.tgz file extension)
- vCenter Server principal file (.json file extension)

Example: Collecting Information About ESXi Hosts and vCenter Server to Be Trusted

This example shows how to use PowerCLI to export the ESXi host information and the vCenter Server Principal. The following table shows the example components and values that are used.

Table 9-5. Example vSphere Trust Authority Setup

Component	Value
ESXi host in Trusted Cluster	192.168.110.51
vCenter Server for Trusted Cluster	192.168.110.22
Variable `$vmhost`	`Get-VMHost`
Variable `$tpm2`	`Get-Tpm2EndorsementKey -VMHost $vmhost`

Table 9-5. Example vSphere Trust Authority Setup (continued)

Component	Value
Trust Authority administrator	trustedadmin@vsphere.local
Local directory to contain output files	C:\vta

```
PS C:\Users\Administrator.CORP> Connect-VIServer -server 192.168.110.51 -User root -Password
'VMware1!'

Name                           Port  User
----                           ----  ----
192.168.110.51                  443  root

PS C:\Users\Administrator.CORP> Get-VMHost

Name               ConnectionState PowerState NumCpu CpuUsageMhz CpuTotalMhz MemoryUsageGB
MemoryTotalGB Version
----               --------------- ---------- ------ ----------- ----------- -------------
------------- -------
192.168.110.51     Connected       PoweredOn       4         200        9576
1.614         7.999   7.0.0

PS C:\Users\Administrator.CORP> $vmhost = Get-VMHost
PS C:\Users\Administrator.CORP> $tpm2 = Get-Tpm2EndorsementKey -VMHost $vmhost
PS C:\> Export-Tpm2CACertificate -Tpm2EndorsementKey $tpm2 -FilePath C:\vta\cacert.zip

Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----        10/8/2019   6:55 PM           1004 cacert.zip

PS C:\Users\Administrator.CORP> Export-VMHostImageDb -VMHost $vmhost -FilePath
C:\vta\image.tgz

Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----         10/8/2019  11:02 PM           2391 image.tgz

PS C:\Users\Administrator.CORP> Disconnect-VIServer -server * -Confirm:$false
PS C:\Users\Administrator.CORP> Connect-VIServer -server 192.168.110.22 -User
trustedadmin@vsphere.local -Password 'VMware1!'

Name                           Port  User
----                           ----  ----
192.168.110.22                  443  VSPHERE.LOCAL\trustedadmin

PS C:\Users\Administrator.CORP> Export-TrustedPrincipal -FilePath C:\vta\principal.json

Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----         10/8/2019  11:14 PM           1873 principal.json
```

What to do next

Continue with Import the Trusted Host Information to the Trust Authority Cluster.

Export and Import a TPM Endorsement Key Certificate

You can export a TPM endorsement key (EK) certificate from an ESXi host, and import it to the vSphere Trust Authority Cluster. You do so when you want to trust an individual ESXi host in the Trusted Cluster.

To import a TPM EK certificate into the Trust Authority Cluster, you must change the Trust Authority Cluster's default attestation type to accept EK certificates. The default attestation type accepts TPM Certificate Authority (CA) certificates. Some TPMs do not include EK certificates. If you want to trust individual ESXi hosts, the TPM must include an EK certificate.

Note Store the exported EK certificate files in a secure location, in case you must restore the vSphere Trust Authority configuration.

Prerequisites

- Enable the Trust Authority Administrator.
- Enable the Trust Authority State.

Procedure

1. Ensure that you are connected as the Trust Authority administrator to the vCenter Server of the Trust Authority Cluster.

   For example, you can enter `$global:defaultviservers` to show all the connected servers.

2. (Optional) If necessary, you can run the following commands to ensure that you are connected to the vCenter Server of the Trust Authority Cluster.

   ```
   Disconnect-VIServer -server * -Confirm:$false
   Connect-VIServer -server TrustAuthorityCluster_VC_ip_address -User trust_admin_user
   -Password 'password'
   ```

3. To change the Trust Authority Cluster's attestation type:

   a. Run the `Get-TrustAuthorityCluster` cmdlet to show the clusters managed by this vCenter Server.

   ```
   Get-TrustAuthorityCluster
   ```

   The clusters are displayed.

   b. Assign the `Get-TrustAuthorityCluster` information to a variable.

   For example, this command assigns the cluster named `vTA Cluster` to the variable `$vTA`.

   ```
   $vTA = Get-TrustAuthorityCluster 'vTA Cluster'
   ```

c Assign the `Get-TrustAuthorityTpm2AttestationSettings` information to a variable.

For example, this command assigns the information to the variable `$tpm2Settings`.

```
$tpm2Settings = Get-TrustAuthorityTpm2AttestationSettings -TrustAuthorityCluster $vTA
```

d Run the `Set-TrustAuthorityTpm2AttestationSettings` cmdlet, specifying `RequireEndorsementKey`, or `RequireCertificateValidation`, or both.

For example, this command specifies `RequireEndorsementKey`.

```
Set-TrustAuthorityTpm2AttestationSettings -Tpm2AttestationSettings $tpm2Settings
-RequireEndorsementKey
```

The system responds with a confirmation prompt similar to the following.

```
Confirmation
Configure the Tpm2AttestationSettings 'TrustAuthorityTpm2AttestationSettings-domain-
c8' with the following parameters:
 RequireCertificateValidation: False
 RequireEndorsementKey: True
[Y] Yes  [A] Yes to All  [N] No  [L] No to All  [S] Suspend  [?] Help (default is "Y"):
```

e At the confirmation prompt, press Enter. (The default is **Y**.)

The output shows a status of True for the setting specified. For example, this status shows True for Require Endorsement Key, and False for Require Certificate Validation.

```
Name                                    RequireEndorsementKey
RequireCertificateValidation   Health
----                                    ---------------------
----------------------------   ------
TrustAuthorityTpm2AttestationSettings... True
False                          Ok
```

4 To export the TPM EK certificate:

a Disconnect from the vCenter Server of the Trust Authority Cluster.

```
Disconnect-VIServer -server * -Confirm:$false
```

b Run the `Connect-VIServer` cmdlet to connect as the root user to one of the ESXi hosts in the Trusted Cluster.

```
Connect-VIServer -server host_ip_address -User root -Password 'password'
```

c Run the `Get-VMHost` cmdlet to confirm the ESXi host.

```
Get-VMHost
```

The host information is displayed.

d Assign `Get-VMHost` to a variable.

For example:

```
$vmhost = Get-VMHost
```

e Run the `Export-Tpm2EndorsementKey` cmdlet to export the EK certificate of the ESXi host.

For example, this command exports the EK certificate to the `tpm2ek.json` file.

```
Export-Tpm2EndorsementKey -VMHost $vmhost -FilePath C:\vta\tpm2ek.json
```

The file is created.

5 To import the TPM EK:

a Disconnect from the ESXi host in the Trusted Cluster.

```
Disconnect-VIServer -server * -Confirm:$false
```

b Connect to the vCenter Server of the Trust Authority Cluster using the Trust Authority administrator user.

```
Connect-VIServer -server TrustAuthorityCluster_VC_ip_address -User trust_admin_user
-Password 'password'
```

c Run the `Get-TrustAuthorityCluster` cmdlet.

```
Get-TrustAuthorityCluster
```

The clusters in the Trust Authority Cluster are displayed.

d Assign the `Get-TrustAuthorityCluster` '*cluster*' information to a variable.

For example, this command assigns the information for cluster `vTA Cluster` to the variable `$vTA`.

```
$vTA = Get-TrustAuthorityCluster 'vTA Cluster'
```

e Run the `New-TrustAuthorityTpm2EndorsementKey` cmdlet.

For example, this command uses the `tpm2ek.json` file previously exported in Step 4.

```
New-TrustAuthorityTpm2EndorsementKey -TrustAuthorityCluster $vTA -FilePath
C:\vta\tpm2ek.json
```

The imported endorsement key information is displayed.

Results

The Trust Authority Cluster's attestation type is changed to accept EK certificates. The EK certificate is exported from the Trusted Cluster and imported to the Trust Authority Cluster.

Example: Export and Import a TPM EK Certificate

This example shows how to use PowerCLI to change the Trust Authority Cluster's default attestation type to accept EK certificates, export the TPM EK certificate from the ESXi host in the Trusted Cluster, and import it to the Trust Authority Cluster. The following table shows the example components and values that are used.

Table 9-6. Example vSphere Trust Authority Setup

Component	Value
vCenter Server for Trust Authority Cluster	192.168.210.22
Variable `$vTA`	`Get-TrustAuthorityCluster 'vTA Cluster'`
Variable `$tpm2Settings`	`Get-TrustAuthorityTpm2AttestationSettings -TrustAuthorityCluster $vTA`
Variable `$vmhost`	`Get-VMHost`
ESXi host in Trusted Cluster	192.168.110.51
Trust Authority administrator	trustedadmin@vsphere.local
Local directory to contain output file	`C:\vta`

```
PS C:\Users\Administrator> Connect-VIServer -server 192.168.210.22 -User
trustedadmin@vsphere.local -Password 'VMware1!'

Name                           Port  User
----                           ----  ----
192.168.210.22                 443   VSPHERE.LOCAL\TrustedAdmin

PS C:\Users\Administrator> Get-TrustAuthorityCluster

Name                 State                Id
----                 -----                --
vTA Cluster          Enabled              TrustAuthorityCluster-domain-c8

PS C:\Users\Administrator> $vTA = Get-TrustAuthorityCluster 'vTA Cluster'

PS C:\Users\Administrator> $tpm2Settings = Get-TrustAuthorityTpm2AttestationSettings
-TrustAuthorityCluster $vTA

PS C:\Users\Administrator> Set-TrustAuthorityTpm2AttestationSettings -Tpm2AttestationSettings
$tpm2Settings -RequireEndorsementKey

Confirmation
Configure the Tpm2AttestationSettings 'TrustAuthorityTpm2AttestationSettings-domain-c8' with
the following parameters:
 RequireCertificateValidation: False
 RequireEndorsementKey: True
[Y] Yes  [A] Yes to All  [N] No  [L] No to All  [S] Suspend  [?] Help (default is "Y"): y

Name                                    RequireEndorsementKey
RequireCertificateValidation   Health
```

```
----                                    ---------------------
---------------------------   ------
TrustAuthorityTpm2AttestationSettings... True
False                         Ok

PS C:\Users\Administrator> Disconnect-VIServer -server * -Confirm:$false
PS C:\Users\Administrator> Connect-VIServer -server 192.168.110.51 -User root -Password
'VMware1!'

Name                          Port  User
----                          ----  ----
192.168.110.51                443   root

PS C:\Users\Administrator> Get-VMHost

Name                 ConnectionState PowerState NumCpu CpuUsageMhz CpuTotalMhz
MemoryUsageGB   MemoryTotalGB Version
----                 --------------- ---------- ------ ----------- -----------
-------------   ------------- -------
192.168.110.51       Connected       PoweredOn       4          55        9576
1.230           7.999   7.0.0

PS C:\Users\Administrator> $vmhost = Get-VMHost
PS C:\Users\Administrator> Export-Tpm2EndorsementKey -VMHost $vmhost -FilePath
C:\vta\tpm2ek.json

Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----        12/3/2019  10:16 PM           2391 tpm2ek.json

PS C:\Users\Administrator> Disconnect-VIServer -server * -Confirm:$false
PS C:\Users\Administrator> Connect-VIServer -server 192.168.210.22 -User
trustedadmin@vsphere.local -Password 'VMware1!'

Name                          Port  User
----                          ----  ----
192.168.210.22                443   VSPHERE.LOCAL\TrustedAdmin

PS C:\Users\Administrator> Get-TrustAuthorityCluster

Name                 State                Id
----                 -----                --
vTA Cluster          Enabled              TrustAuthorityCluster-domain-c8

PS C:\Users\Administrator> $vTA = Get-TrustAuthorityCluster ‘vTA Cluster’
PS C:\Users\Administrator> New-TrustAuthorityTpm2EndorsementKey -TrustAuthorityCluster $vTA
-FilePath C:\vta\tpm2ek.json

TrustAuthorityClusterId                 Name                                    Health
-----------------------                 ----                                    ------
TrustAuthorityCluster-domain-c8         1a520e42-4db8-1cbb-6dd7-f493fd921ccb    Ok
```

What to do next

Continue with Import the Trusted Host Information to the Trust Authority Cluster.

Import the Trusted Host Information to the Trust Authority Cluster

You import the exported ESXi host and vCenter Server information into the vSphere Trust Authority Cluster, so that the Trust Authority Cluster knows which hosts it can attest.

If you are following these tasks in order, you are still connected to the vCenter Server of the Trust Authority Cluster.

Prerequisites

- Enable the Trust Authority Administrator.
- Enable the Trust Authority State.
- Collect Information About ESXi Hosts and vCenter Server to Be Trusted.

Procedure

1. Ensure that you are connected as the Trust Authority administrator to the vCenter Server of the Trust Authority Cluster.

   For example, you can enter `$global:defaultviservers` to show all the connected servers.

2. (Optional) If necessary, you can run the following commands to ensure that you are connected to the vCenter Server of the Trust Authority Cluster.

   ```
   Disconnect-VIServer -server * -Confirm:$false
   Connect-VIServer -server TrustAuthorityCluster_VC_ip_address -User trust_admin_user
   -Password 'password'
   ```

3. To show the clusters managed by this vCenter Server, run the `Get-TrustAuthorityCluster` cmdlet.

   ```
   Get-TrustAuthorityCluster
   ```

   The clusters are displayed.

4. Assign the `Get-TrustAuthorityCluster` '*cluster*' information to a variable.

   For example, this command assigns the information for cluster `vTA Cluster` to the variable `$vTA`.

   ```
   $vTA = Get-TrustAuthorityCluster 'vTA Cluster'
   ```

5. To import the vCenter Server principal information of the Trusted Cluster into the Trust Authority Cluster, run the `New-TrustAuthorityPrincipal` cmdlet.

   For example, the following command imports the `principal.json` file previously exported in Collect Information About ESXi Hosts and vCenter Server to Be Trusted.

   ```
   New-TrustAuthorityPrincipal -TrustAuthorityCluster $vTA -FilePath C:\vta\principal.json
   ```

   The TrustAuthorityPrincipal information is displayed.

6 To verify the import, run the `Get-TrustAuthorityPrincipal` cmdlet.

For example:

```
Get-TrustAuthorityPrincipal -TrustAuthorityCluster $vTA
```

The imported TrustAuthorityPrincipal information is displayed.

7 To import the Trusted Platform Module (TPM) CA certificate information, run the `New-TrustAuthorityTpm2CACertificate` cmdlet.

For example, the following command imports the TPM CA certificate information from the `cacert.zip` file previously exported in Collect Information About ESXi Hosts and vCenter Server to Be Trusted.

```
New-TrustAuthorityTpm2CACertificate -TrustAuthorityCluster $vTA -FilePath
C:\vta\cacert.zip
```

The imported certificate information is displayed.

8 To import the ESXi host base image information, run the `New-TrustAuthorityVMHostBaseImage` cmdlet.

For example, the following command imports the image information from the `image.tgz` file previously exported in Collect Information About ESXi Hosts and vCenter Server to Be Trusted.

```
New-TrustAuthorityVMHostBaseImage -TrustAuthorityCluster $vTA -FilePath C:\vta\image.tgz
```

The imported image information is displayed.

Results

The Trust Authority Cluster knows which ESXi hosts it can remotely attest, and so, which hosts it can trust.

Example: Import the Trusted Host Information to the Trust Authority Cluster

This example shows how to use PowerCLI to import the vCenter Server principal information of the Trusted Cluster and the Trusted Host information files to the Trust Authority Cluster. It assumes that you are connected to the vCenter Server of the Trust Authority Cluster as the Trust Authority administrator. The following table shows the example components and values that are used.

Table 9-7. Example vSphere Trust Authority Setup

Component	Value
Variable `$vTA`	`Get-TrustAuthorityCluster 'vTA Cluster1'`
vCenter Server for Trust Authority Cluster	192.168.210.22

Table 9-7. Example vSphere Trust Authority Setup (continued)

Component	Value
Trust Authority Cluster names	vTA Cluster1 (Enabled) vTA Cluster2 (Disabled)
Principal information file	`C:\vta\principal.json`
TPM certificate file	`C:\vta\cacert.cer`
ESXi host base image file	`C:\vta\image.tgz`
Trust Authority administrator	trustedadmin@vsphere.local

```
PS C:\Users\Administrator> Disconnect-VIServer -server * -Confirm:$false
PS C:\Users\Administrator> Connect-VIServer -server 192.168.210.22 -User
trustedadmin@vsphere.local -Password 'VMware1!'

Name                           Port  User
----                           ----  ----
192.168.210.22                  443  VSPHERE.LOCAL\trustedadmin

PS C:\Users\Administrator> Get-TrustAuthorityCluster

Name                 State                Id
----                 -----                --
vTA Cluster1         Enabled              TrustAuthorityCluster-domain-c8
vTA Cluster2         Disabled             TrustAuthorityCluster-domain-c26

PS C:\Users\Administrator> $vTA = Get-TrustAuthorityCluster 'vTA Cluster1'

PS C:\Users\Administrator.CORP> New-TrustAuthorityPrincipal -TrustAuthorityCluster $vTA
-FilePath C:\vta\principal.json

Name                                          Domain          Type
TrustAuthorityClusterId
----                                          ------          ----
-----------------------
vpxd-de207929-0601-43ef-9616-47d0cee0302f     vsphere.local   STS_USER
TrustAuthorityCluster-domain-c8

PS C:\Users\Administrator.CORP> Get-TrustAuthorityPrincipal -TrustAuthorityCluster $vTA

Name                                          Domain          Type
TrustAuthorityClusterId
----                                          ------          ----
-----------------------
vpxd-de207929-0601-43ef-9616-47d0cee0302f     vsphere.local   STS_USER
TrustAuthorityCluster-domain-c8

PS C:\Users\Administrator.CORP> New-TrustAuthorityTpm2CACertificate -TrustAuthorityCluster
$vTA -FilePath C:\vta\cacert.cer

TrustAuthorityClusterId                 Name                                        Health
```

```
-----------------------                    ----                                       ------
TrustAuthorityCluster-domain-c8            52BDB7B4B2F55C925C047257DED4588A7767D961 Ok

PS C:\Users\Administrator.CORP> New-TrustAuthorityVMHostBaseImage -TrustAuthorityCluster $vTA
-FilePath C:\vta\image.tgz

TrustAuthorityClusterId                    VMHostVersion                              Health
-----------------------                    -------------                              ------
TrustAuthorityCluster-domain-c8            ESXi 7.0.0-0.0.14828939                    Ok
```

What to do next

Continue with Create the Key Provider on the Trust Authority Cluster.

Create the Key Provider on the Trust Authority Cluster

For the Key Provider Service to connect to a key provider, you must create a trusted key provider then configure a trust setup between the vSphere Trust Authority Cluster and the key server (KMS). For most KMIP-compliant key servers, this configuration involves setting up client and server certificates.

What was previously called a KMS Cluster in vSphere 6.7 is now called a key provider in vSphere 7.0. For more information about key providers, see What Is the vSphere Trust Authority Key Provider Service.

In a production environment, you can create multiple key providers. By creating multiple key providers, you can address how to manage your deployment based on company organization, different business units or customers, and so on.

If you are following these tasks in order, you are still connected to the vCenter Server of the vSphere Trust Authority Cluster.

Prerequisites

- Enable the Trust Authority Administrator.
- Enable the Trust Authority State.
- Collect Information About ESXi Hosts and vCenter Server to Be Trusted.
- Import the Trusted Host Information to the Trust Authority Cluster.
- Create and activate a key on the key server to be the primary key for the trusted key provider. This key wraps other keys and secrets used by this trusted key provider. See your key server vendor documentation for more information about creating keys.

Procedure

1 Ensure that you are connected to the vCenter Server of the Trust Authority Cluster. For example, you can enter `$global:defaultviservers` to show all the connected servers.

2 (Optional) If necessary, you can run the following commands to ensure that you are connected to the vCenter Server of the Trust Authority Cluster.

```
Disconnect-VIServer -server * -Confirm:$false
Connect-VIServer -server TrustAuthorityCluster_VC_ip_address -User trust_admin_user
-Password 'password'
```

3 To create the trusted key provider, run the `New-TrustAuthorityKeyProvider` cmdlet.

For example, this command uses `1` for the PrimaryKeyID and the name `clkp`. If you are following these tasks in order, you previously assigned `Get-TrustAuthorityCluster` information to a variable (for example, `$vTA = Get-TrustAuthorityCluster 'vTA Cluster'`).

```
New-TrustAuthorityKeyProvider -TrustAuthorityCluster $vTA -PrimaryKeyId 1 -Name clkp
-KmipServerAddress ip_address
```

The PrimaryKeyID is normally a key ID that comes from the key server in the form of a UUID. Do not use the key name for PrimaryKeyID. The PrimaryKeyID value is vendor-dependent. See your key server documentation. The `New-TrustAuthorityKeyProvider` cmdlet can take other options, such as `KmipServerPort`, `ProxyAddress`, and `ProxyPort`. See the `New-TrustAuthorityKeyProvider` Help system for more information.

Each logical key provider, regardless of its type (Standard, Trusted, and Native Key Provider), must have a unique name across all vCenter Server systems.

For more information, see Key Provider Naming.

Note To add multiple key servers to the key provider, use the `Add-TrustAuthorityKeyProviderServer` cmdlet.

The key provider information is displayed.

4 Establish the trusted connection so that the key server trusts the trusted key provider. The exact process depends on the certificates that the key server accepts, and on your company policy. Select the option appropriate for your server and finish the steps.

Option	See
Upload Client Certificate	Upload the Client Certificate to Establish a Trusted Key Provider Trusted Connection.
Upload KMS certificate and private key	Upload the Certificate and Private Key to Establish a Trusted Key Provider Trusted Connection.
New Certificate Signing Request	Create a Certificate Signing Request to Establish a Trusted Key Provider Trusted Connection.

5 Finish the trust setup by uploading a key server certificate so that the trusted key provider trusts the key server.

a Assign the `Get-TrustAuthorityKeyProvider -TrustAuthorityCluster $vTA` information to a variable.

For example:

```
$kp = Get-TrustAuthorityKeyProvider -TrustAuthorityCluster $vTA
```

This variable obtains the trusted key providers in the given Trust Authority Cluster, in this case, `$vTA`.

Note If you have more than one trusted key provider, use commands similar to the following to select the one you want:

```
Get-TrustAuthorityKeyProvider -TrustAuthorityCluster $vTA
<The trusted key providers listing is displayed.>
$kp = Get-TrustAuthorityKeyProvider -TrustAuthorityCluster $vTA | Select-Object -Last 1
```

Using `Select-Object -Last 1` selects the last trusted key provider in the list.

b To get the key server server certificate, run the `Get-TrustAuthorityKeyProviderServerCertificate` command.

For example:

```
Get-TrustAuthorityKeyProviderServerCertificate -KeyProviderServer
$kp.KeyProviderServers
```

The server certificate information is displayed. Initially, the certificate is not trusted, so the Trusted state is False. If you have more than one key server configured, a list of certificates is returned. Verify and add each certificate using the following instructions.

c Before trusting the certificate, assign `Get-TrustAuthorityKeyProviderServerCertificate -KeyProviderServer $kp.KeyProviderServers` information to a variable (for example, `cert`), and run the `$cert.Certificate.ToString()` command and verify the output.

For example:

```
$cert = Get-TrustAuthorityKeyProviderServerCertificate -KeyProviderServer
$kp.KeyProviderServers
$cert.Certificate.ToString()
```

The certificate information is displayed, including Subject, Issuer, and other information.

d To add the KMIP server certificate to the trusted key provider, run `Add-TrustAuthorityKeyProviderServerCertificate`.

For example:

```
Add-TrustAuthorityKeyProviderServerCertificate -ServerCertificate $cert
```

The certificate information is displayed and the Trusted state is now True.

6 Verify the status of the key provider.

a To refresh the key provider status, reassign the `$kp` variable.

For example:

```
$kp = Get-TrustAuthorityKeyProvider -TrustAuthorityCluster $vTA
```

Note If you have more than one trusted key provider, use commands similar to the following to select the one you want:

```
Get-TrustAuthorityKeyProvider -TrustAuthorityCluster $vTA
<The trusted key providers listing is displayed.>
$kp = Get-TrustAuthorityKeyProvider -TrustAuthorityCluster $vTA | Select-Object -Last 1
```

Using `Select-Object -Last 1` selects the last trusted key provider in the list.

b Run the `$kp.Status` command to get the key provider status.

For example:

```
$kp.Status
```

Note The status can take a few minutes to be refreshed. To view the status, reassign the `$kp` variable and rerun the `$kp.Status` command.

A Health status of Ok indicates that the key provider is running correctly.

Results

The trusted key provider is created and has established trust with the key server.

Example: Create the Key Provider on the Trust Authority Cluster

This example shows how to use PowerCLI to create the trusted key provider on the Trust Authority Cluster. It assumes that you are connected to the vCenter Server of the Trust Authority Cluster as the Trust Authority administrator. It also uses a certificate signed by the key server vendor after submitting a CSR to the vendor.

The following table shows the example components and values that are used.

Table 9-8. Example vSphere Trust Authority Setup

Component	Value
Variable `$vTA`	`Get-TrustAuthorityCluster 'vTA Cluster'`
Variable `$kp`	`Get-TrustAuthorityKeyProvider -TrustAuthorityCluster $vTA`
Variable `$cert`	`Get-TrustAuthorityKeyProviderServerCertificate -KeyProviderServer $kp.KeyProviderServers`
vCenter Server for Trust Authority Cluster	192.168.210.22
KMIP-compliant key server	192.168.110.91
KMIP-compliant key server user	vcqekmip
Trust Authority Cluster name	vTA Cluster
Trust Authority administrator	trustedadmin@vsphere.local

```
PS C:\Users\Administrator.CORP> Disconnect-VIServer -server * -Confirm:$false
PS C:\Users\Administrator.CORP> Connect-VIServer -server 192.168.210.22 -User
trustedadmin@vsphere.local -Password 'VMware1!'

PS C:\Users\Administrator.CORP> New-TrustAuthorityKeyProvider -TrustAuthorityCluster $vTA
-PrimaryKeyId 8 -Name clkp -KmipServerAddress 192.168.110.91
Name                 PrimaryKeyId         Type       TrustAuthorityClusterId
----                 ------------         ----       -----------------------
clkp                 8                    KMIP       TrustAuthorityCluster-domain-c8

PS C:\Users\Administrator.CORP> New-TrustAuthorityKeyProviderClientCertificate -KeyProvider
$kp
<Export the client certificate when you need to use it.>
PS C:\Users\Administrator.CORP> Export-TrustAuthorityKeyProviderClientCertificate
-KeyProvider $kp -FilePath clientcert.pem

PS C:\Users\Administrator.CORP> $kp = Get-TrustAuthorityKeyProvider -TrustAuthorityCluster
$vTA
PS C:\Users\Administrator.CORP> Get-TrustAuthorityKeyProviderServerCertificate
-KeyProviderServer $kp.KeyProviderServers

Certificate                              Trusted    KeyProviderServerId       KeyProviderId
-----------                              -------    -------------------       -------------
[Subject]...                             False      domain-c8-clkp:192.16.... domain-c8-clkp
```

```
PS C:\WINDOWS\system32> $cert.Certificate.ToString()
[Subject]
  E=<domain>, CN=<IP address>, OU=VMware Engineering, O=VMware, L=Palo Alto, S=California,
C=US

[Issuer]
  O=<host>.eng.vmware.com, C=US, DC=local, DC=vsphere, CN=CA

[Serial Number]
  00CEF192BBF9D80C9F

[Not Before]
  8/10/2015 4:16:12 PM

[Not After]
  8/9/2020 4:16:12 PM

[Thumbprint]
  C44068C124C057A3D07F51DCF18720E963604B70

PS C:\Users\Administrator.CORP> $cert = Get-TrustAuthorityKeyProviderServerCertificate
-KeyProviderServer $kp.KeyProviderServers
PS C:\Users\Administrator.CORP> Add-TrustAuthorityKeyProviderServerCertificate
-ServerCertificate $cert

Certificate                              Trusted    KeyProviderServerId       KeyProviderId
-----------                              -------    -------------------       -------------
[Subject]...                             True                                 domain-c8-clkp

PS C:\Users\Administrator.CORP> $kp = Get-TrustAuthorityKeyProvider -TrustAuthorityCluster
$vTA
PS C:\Users\Administrator.CORP> $kp.Status

KeyProviderId Health HealthDetails ServerStatus
------------- ------ ------------- ------------
domain-c8-kp4     Ok {}            {192.168.210.22}
```

What to do next

Continue with Export the Trust Authority Cluster Information.

Upload the Client Certificate to Establish a Trusted Key Provider Trusted Connection

Some key server (KMS) vendors require that you upload the trusted key provider's client certificate to the key server. After the upload, the key server accepts traffic that comes from the trusted key provider.

Prerequisites

- Enable the Trust Authority Administrator.
- Enable the Trust Authority State.
- Collect Information About ESXi Hosts and vCenter Server to Be Trusted.

- Import the Trusted Host Information to the Trust Authority Cluster.
- Create the Key Provider on the Trust Authority Cluster.

Procedure

1. Ensure that you are connected to the vCenter Server of the Trust Authority Cluster. For example, you can enter `$global:defaultviservers` to show all the connected servers.
2. (Optional) If necessary, you can run the following commands to ensure that you are connected to the vCenter Server of the Trust Authority Cluster.

   ```
   Disconnect-VIServer -server * -Confirm:$false
   Connect-VIServer -server TrustAuthorityCluster_VC_ip_address -User trust_admin_user
   -Password 'password'
   ```

3. Assign the `Get-TrustAuthorityKeyProvider -TrustAuthorityCluster $vTA` information to a variable.

   For example:

   ```
   $kp = Get-TrustAuthorityKeyProvider -TrustAuthorityCluster $vTA
   ```

   If you are following these tasks in order, you previously assigned `Get-TrustAuthorityCluster` information to a variable (for example, `$vTA = Get-TrustAuthorityCluster 'vTA Cluster'`).

   This variable obtains the trusted key providers in the given Trust Authority Cluster, in this case, `$vTA`.

   Note If you have more than one trusted key provider, use commands similar to the following to select the one you want:

   ```
   Get-TrustAuthorityKeyProvider -TrustAuthorityCluster $vTA
   <The trusted key providers listing is displayed.>
   $kp = Get-TrustAuthorityKeyProvider -TrustAuthorityCluster $vTA | Select-Object -Last 1
   ```

   Using `Select-Object -Last 1` selects the last trusted key provider in the list.

4. To create the trusted key provider client certificate, run the `New-TrustAuthorityKeyProviderClientCertificate` cmdlet.

   For example:

   ```
   New-TrustAuthorityKeyProviderClientCertificate -KeyProvider $kp
   ```

   The thumbprint is displayed.

5 To export the key provider client certificate, run the `Export-TrustAuthorityKeyProviderClientCertificate` cmdlet.

For example:

```
Export-TrustAuthorityKeyProviderClientCertificate -KeyProvider $kp -FilePath clientcert.pem
```

The certificate is exported to a file.

6 Upload the certificate file to the key server.

See your key server documentation for more information.

Results

The trusted key provider has established trust with the key server.

Upload the Certificate and Private Key to Establish a Trusted Key Provider Trusted Connection

Some key server (KMS) vendors require that you configure the trusted key provider with the client certificate and private key provided by the key server. After you configure the trusted key provider, the key server accepts traffic from the trusted key provider.

Prerequisites

- Enable the Trust Authority Administrator.
- Enable the Trust Authority State.
- Collect Information About ESXi Hosts and vCenter Server to Be Trusted.
- Import the Trusted Host Information to the Trust Authority Cluster.
- Create the Key Provider on the Trust Authority Cluster.
- Request a certificate and private key in PEM format from the key server vendor. If the certificate is returned in a format other than PEM, convert it to PEM. If the private key is protected with a password, create a PEM file with the password removed. You can use the `openssl` command for both operations. For example:
  - To convert a certificate from CRT to PEM format:

    ```
    openssl x509 -in clientcert.crt -out clientcert.pem -outform PEM
    ```

  - To convert a certificate from DER to PEM format:

    ```
    openssl x509 -inform DER -in clientcert.der -out clientcert.pem
    ```

  - To remove the password from a private key:

    ```
    openssl rsa -in key.pem -out keynopassword.pem
    Enter pass phrase for key.pem:
    writing RSA key
    ```

Procedure

1 Ensure that you are connected to the vCenter Server of the Trust Authority Cluster. For example, you can enter `$global:defaultviservers` to show all the connected servers.

2 (Optional) If necessary, you can run the following commands to ensure that you are connected to the vCenter Server of the Trust Authority Cluster.

```
Disconnect-VIServer -server * -Confirm:$false
Connect-VIServer -server TrustAuthorityCluster_VC_ip_address -User trust_admin_user
-Password 'password'
```

3 Assign the `Get-TrustAuthorityKeyProvider -TrustAuthorityCluster $vTA` information to a variable.

For example:

```
$kp = Get-TrustAuthorityKeyProvider -TrustAuthorityCluster $vTA
```

If you are following these tasks in order, you previously assigned `Get-TrustAuthorityCluster` information to a variable (for example, `$vTA = Get-TrustAuthorityCluster 'vTA Cluster'`).

The `$kp` variable obtains the trusted key providers in the given Trust Authority Cluster, in this case, `$vTA`.

Note If you have more than one trusted key provider, use commands similar to the following to select the one you want:

```
Get-TrustAuthorityKeyProvider -TrustAuthorityCluster $vTA
<The trusted key providers listing is displayed.>
$kp = Get-TrustAuthorityKeyProvider -TrustAuthorityCluster $vTA | Select-Object -Last 1
```

Using `Select-Object -Last 1` selects the last trusted key provider in the list.

4 Upload the certificate and private key using the `Set-TrustAuthorityKeyProviderClientCertificate` command.

For example:

```
Set-TrustAuthorityKeyProviderClientCertificate -KeyProvider $kp -CertificateFilePath
<path/to/certfile.pem> -PrivateKeyFilePath <path/to/privatekey.pem>
```

Results

The trusted key provider has established trust with the key server.

Create a Certificate Signing Request to Establish a Trusted Key Provider Trusted Connection

Some key server (KMS) vendors require that you generate a Certificate Signing Request (CSR) and send that CSR to the key server vendor. The key server vendor signs the CSR and returns

the signed certificate. After you configure this signed certificate as the trusted key provider's client certificate, the key server accepts traffic that comes from the trusted key provider.

This task is a two-step process. First you generate the CSR and send it to your key server vendor. Then you upload the signed certificate that you receive from the key server vendor.

Prerequisites

- Enable the Trust Authority Administrator.
- Enable the Trust Authority State.
- Collect Information About ESXi Hosts and vCenter Server to Be Trusted.
- Import the Trusted Host Information to the Trust Authority Cluster.
- Create the Key Provider on the Trust Authority Cluster.

Procedure

1. Ensure that you are connected to the vCenter Server of the Trust Authority Cluster. For example, you can enter `$global:defaultviservers` to show all the connected servers.
2. (Optional) If necessary, you can run the following commands to ensure that you are connected to the vCenter Server of the Trust Authority Cluster.

   ```
   Disconnect-VIServer -server * -Confirm:$false
   Connect-VIServer -server TrustAuthorityCluster_VC_ip_address -User trust_admin_user
   -Password 'password'
   ```

3. Assign the `Get-TrustAuthorityKeyProvider -TrustAuthorityCluster $vTA` information to a variable.

   For example:

   ```
   $kp = Get-TrustAuthorityKeyProvider -TrustAuthorityCluster $vTA
   ```

   If you are following these tasks in order, you previously assigned `Get-TrustAuthorityCluster` information to a variable (for example, `$vTA = Get-TrustAuthorityCluster 'vTA Cluster'`).

   This variable obtains the trusted key providers in the given Trust Authority Cluster, in this case, `$vTA`.

   Note If you have more than one trusted key provider, use commands similar to the following to select the one you want:

   ```
   Get-TrustAuthorityKeyProvider -TrustAuthorityCluster $vTA
   <The trusted key providers listing is displayed.>
   $kp = Get-TrustAuthorityKeyProvider -TrustAuthorityCluster $vTA | Select-Object -Last 1
   ```

   Using `Select-Object -Last 1` selects the last trusted key provider in the list.

4 To generate a CSR, use the `New-TrustAuthorityKeyProviderClientCertificateCSR` cmdlet.

For example:

```
New-TrustAuthorityKeyProviderClientCertificateCSR -KeyProvider $kp
```

The CSR is displayed. You can also use the `Get-TrustAuthorityKeyProviderClientCertificateCSR -KeyProvider $kp` cmdlet to obtain the CSR.

5 To get a signed certificate, submit the CSR to your key server vendor.

The certificate must be in PEM format. If the certificate is returned in a format other than PEM, convert it to PEM by using the `openssl` command. For example:

- To convert a certificate from CRT to PEM format:

```
openssl x509 -in clientcert.crt -out clientcert.pem -outform PEM
```

- To convert a certificate from DER to PEM format:

```
openssl x509 -inform DER -in clientcert.der -out clientcert.pem
```

6 When you receive the signed certificate from the key server vendor, upload the certificate to the key server using the `Set-TrustAuthorityKeyProviderClientCertificate` cmdlet.

For example:

```
Set-TrustAuthorityKeyProviderClientCertificate -KeyProvider $kp -CertificateFilePath
<path/tp/certfile.pem>
```

Results

The trusted key provider has established trust with the key server.

Export the Trust Authority Cluster Information

For the Trusted Cluster to connect to the vSphere Trust Authority Cluster, you must export the Trust Authority Cluster's service information in the form of a file then import that file to the Trusted Cluster. You must ensure to keep this file confidential and transport it securely.

If you are following these tasks in order, you are still connected to the vCenter Server of the Trust Authority Cluster.

Note Store the exported service information file in a secure location, in case you must restore the vSphere Trust Authority configuration.

Prerequisites

- Enable the Trust Authority Administrator.

- Enable the Trust Authority State.
- Collect Information About ESXi Hosts and vCenter Server to Be Trusted.
- Import the Trusted Host Information to the Trust Authority Cluster.
- Create the Key Provider on the Trust Authority Cluster.

Procedure

1. Ensure that you are connected to the vCenter Server of the Trust Authority Cluster. For example, you can enter `$global:defaultviservers` to show all the connected servers.
2. (Optional) If necessary, you can run the following commands to ensure that you are connected to the vCenter Server of the Trust Authority Cluster.

   ```
   Disconnect-VIServer -server * -Confirm:$false
   Connect-VIServer -server TrustAuthorityCluster_VC_ip_address -User trust_admin_user
   -Password 'password'
   ```

3. To export the Trust Authority Cluster's Attestation Service and Key Provider Service information, run the `Export-TrustAuthorityServicesInfo` cmdlet.

   For example, this command exports the service information to the `clsettings.json` file. If you are following these tasks in order, you previously assigned the `Get-TrustAuthorityCluster` information to a variable (for example, `$vTA = Get-TrustAuthorityCluster 'vTA Cluster'`).

   ```
   Export-TrustAuthorityServicesInfo -TrustAuthorityCluster $vTA -FilePath
   C:\vta\clsettings.json
   ```

   The file is created.

Results

A file containing the Trust Authority Cluster information is created.

Example: Export the Trust Authority Cluster Information

This example shows how to use PowerCLI to export the Trust Authority Cluster service information. The following table shows the example components and values that are used.

Table 9-9. Example vSphere Trust Authority Setup

Component	Value
Variable `$vTA`	`Get-TrustAuthorityCluster 'vTA Cluster'`
vCenter Server for Trust Authority Cluster	192.168.210.22
Trust Authority administrator	trustedadmin@vsphere.local

```
PS C:\Users\Administrator.CORP> Disconnect-VIServer -server * -Confirm:$false
PS C:\Users\Administrator.CORP> Connect-VIServer -server 192.168.210.22 -User
```

```
trustedadmin@vsphere.local -Password 'VMware1!'

PS C:\Users\Administrator.CORP> Export-TrustAuthorityServicesInfo -TrustAuthorityCluster $vTA
-FilePath C:\vta\clsettings.json

Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----       10/16/2019   9:59 PM           8177 clsettings.json
```

What to do next

Continue with Import the Trust Authority Cluster Information to the Trusted Hosts.

Import the Trust Authority Cluster Information to the Trusted Hosts

After you have imported the vSphere Trust Authority Cluster information to the Trusted Cluster, the Trusted Hosts start the attestation process with the Trust Authority Cluster.

Prerequisites

- Enable the Trust Authority Administrator.
- Enable the Trust Authority State.
- Collect Information About ESXi Hosts and vCenter Server to Be Trusted.
- Import the Trusted Host Information to the Trust Authority Cluster.
- Create the Key Provider on the Trust Authority Cluster.
- Export the Trust Authority Cluster Information.

Procedure

1. Ensure that you are connected as the Trust Authority administrator to the vCenter Server of the Trusted Cluster.

   For example, you can enter `$global:defaultviservers` to show all the connected servers.

2. (Optional) If necessary, you can run the following commands to ensure that you are connected to the vCenter Server of the Trusted Cluster.

   ```
   Disconnect-VIServer -server * -Confirm:$false
   Connect-VIServer -server TrustedCluster_VC_ip_address -User trust_admin_user -Password
   'password'
   ```

   Note Alternatively, you can start another PowerCLI session to connect to the vCenter Server of the Trusted Cluster.

3. Verify that the state of the Trusted Cluster is disabled.

   ```
   Get-TrustedCluster
   ```

   The State is shown as Disabled.

4 Assign the `Get-TrustedCluster` information to a variable.

For example, this command assigns information for the cluster `Trusted Cluster` to the variable `$TC`.

```
$TC = Get-TrustedCluster -Name 'Trusted Cluster'
```

5 Verify the value of the variable by echoing it.

For example:

```
$TC
```

The `Get-TrustedCluster` information is displayed.

6 To import the Trust Authority Cluster information to the vCenter Server, run the `Import-TrustAuthorityServicesInfo` cmdlet.

For example, this command imports the service information from the `clsettings.json` file previously exported in Export the Trust Authority Cluster Information.

```
Import-TrustAuthorityServicesInfo -FilePath C:\vta\clsettings.json
```

The system responds with a confirmation prompt.

```
Confirmation
Importing the TrustAuthorityServicesInfo into Server 'ip_address'. Do you want to proceed?

[Y] Yes  [A] Yes to All  [N] No  [L] No to All  [S] Suspend  [?] Help (default is "Y"):
```

7 At the confirmation prompt, press Enter. (The default is **Y**.)

The service information for the hosts in the Trust Authority Cluster is displayed.

8 To enable the Trusted Cluster, run the `Set-TrustedCluster` cmdlet.

For example:

```
Set-TrustedCluster -TrustedCluster $TC -State Enabled
```

The system responds with a confirmation prompt.

```
Confirmation
Setting TrustedCluster 'cluster' with new TrustedState 'Enabled'. Do you want to proceed?
[Y] Yes  [A] Yes to All  [N] No  [L] No to All  [S] Suspend  [?] Help (default is "Y"):
```

If the Trusted Cluster is not in a healthy state, the following warning message is displayed before the confirmation message:

```
WARNING: The TrustedCluster 'cluster' is not healthy in its TrustedClusterAppliedStatus.
This cmdlet will automatically remediate the TrustedCluster.
```

9 At the confirmation prompt, press Enter. (The default is `Y`.)

The Trusted Cluster is enabled.

Note You can also enable the Trusted Cluster by enabling the Attestation Service and the Key Provider Service individually. Use the `Add-TrustedClusterAttestationServiceInfo` and `Add-TrustedClusterKeyProviderServiceInfo` commands. For example, the following commands enable the services one at a time for the cluster `Trusted Cluster` that has two Key Provider Services and two Attestation Services.

```
Add-TrustedClusterAttestationServiceInfo -TrustedCluster 'Trusted Cluster'
-AttestationServiceInfo (Get-AttestationServiceInfo | Select-Object -index 0,1)
Add-TrustedClusterKeyProviderServiceInfo  -TrustedCluster 'Trusted Cluster'
-KeyProviderServiceInfo (Get-KeyProviderServiceInfo | Select-Object -index 0,1)
```

10 Verify that the Attestation Service and the Key Provider Service are configured in the Trusted Cluster.

a Assign the `Get-TrustedCluster` information to a variable.

For example, this command assigns information for the cluster `Trusted Cluster` to the variable `$TC`.

```
$TC = Get-TrustedCluster -Name 'Trusted Cluster'
```

b Verify that the Attestation Service is configured.

```
$tc.AttestationServiceInfo
```

The Attestation Service information is displayed.

c Verify that the Key Provider Service is configured.

```
$tc.KeyProviderServiceInfo
```

The Key Provider Service information is displayed.

Results

The ESXi Trusted Hosts in the Trusted Cluster begin the attestation process with the Trust Authority Cluster.

Example: Import the Trust Authority Cluster Information to the Trusted Hosts

This example shows how to import the Trust Authority Cluster service information to the Trusted Cluster. The following table shows the example components and values that are used.

Table 9-10. Example vSphere Trust Authority Setup

Component	Value
vCenter Server of the Trusted Cluster	192.168.110.22
Trust Authority administrator	trustedadmin@vsphere.local
Trusted Cluster name	Trusted Cluster
ESXi hosts in the Trust Authority Cluster	192.168.210.51 and 192.168.210.52
Variable `$TC`	`Get-TrustedCluster -Name 'Trusted Cluster'`

```
PS C:\Users\Administrator.CORP> Disconnect-VIServer -server * -Confirm:$false
PS C:\Users\Administrator.CORP> Connect-VIServer -server 192.168.110.22 -User
trustedadmin@vsphere.local -Password 'VMware1!'

Name                           Port  User
----                           ----  ----
192.168.110.22                 443   VSPHERE.LOCAL\trustedadmin

PS C:\Users\Administrator.CORP> Get-TrustedCluster

Name                  State             Id
----                  -----             --
Trusted Cluster       Disabled          TrustedCluster-domain-c8

PS C:\Users\Administrator.CORP> $TC = Get-TrustedCluster -Name 'Trusted Cluster'
PS C:\Users\Administrator.CORP> $TC

Name                  State             Id
----                  -----             --
Trusted Cluster       Disabled          TrustedCluster-domain-c8

PS C:\Users\Administrator.CORP> Import-TrustAuthorityServicesInfo -FilePath
C:\vta\clsettings.json

Confirmation
Importing the TrustAuthorityServicesInfo into Server '192.168.110.22'. Do you want to proceed?
[Y] Yes  [A] Yes to All  [N] No  [L] No to All  [S] Suspend  [?] Help (default is "Y"): y

ServiceAddress                 ServicePort          ServiceGroup
--------------                 -----------          ------------
192.168.210.51                 443                  host-13:86f7ab6c-ad6f-4606-...
192.168.210.52                 443                  host-16:86f7ab6c-ad6f-4606-...
192.168.210.51                 443                  host-13:86f7ab6c-ad6f-4606-...
192.168.210.52                 443                  host-16:86f7ab6c-ad6f-4606-...

PS C:\Users\Administrator.CORP> Set-TrustedCluster -TrustedCluster $TC -State Enabled

Confirmation
Setting TrustedCluster 'Trusted Cluster' with new TrustedState 'Enabled'. Do you want to
proceed?
[Y] Yes  [A] Yes to All  [N] No  [L] No to All  [S] Suspend  [?] Help (default is "Y"):
```

```
Name                  State             Id
----                  -----             --
Trusted Cluster       Enabled           TrustedCluster-domain-c8

PS C:\Users\Administrator.CORP> $TC = Get-TrustedCluster -Name 'Trusted Cluster'
PS C:\Users\Administrator.CORP> $tc.AttestationServiceInfo

ServiceAddress                ServicePort          ServiceGroup
--------------                -----------          ------------
192.168.210.51                443                  host-13:dc825986-73d2-463c-...
192.168.210.52                443                  host-16:dc825986-73d2-463c-...

PS C:\Users\Administrator.CORP> $tc.KeyProviderServiceInfo

ServiceAddress                ServicePort          ServiceGroup
--------------                -----------          ------------
192.168.210.51                443                  host-13:dc825986-73d2-463c-...
192.168.210.52                443                  host-16:dc825986-73d2-463c-...
```

What to do next

Continue with Configure the Trusted Key Provider for Trusted Hosts Using the vSphere Client or Configure the Trusted Key Provider for Trusted Hosts Using the Command Line.

Configure the Trusted Key Provider for Trusted Hosts Using the vSphere Client

You can configure the trusted key provider by using the vSphere Client.

Prerequisites

- Enable the Trust Authority Administrator.
- Enable the Trust Authority State.
- Collect Information About ESXi Hosts and vCenter Server to Be Trusted.
- Import the Trusted Host Information to the Trust Authority Cluster.
- Create the Key Provider on the Trust Authority Cluster.
- Export the Trust Authority Cluster Information.
- Import the Trust Authority Cluster Information to the Trusted Hosts.

Procedure

1. Connect to vCenter Server of the Trusted Cluster by using the vSphere Client.
2. Log in as the vCenter Server administrator, or an administrator that has the **Cryptographic operations.Manage key servers** privilege.
3. Select the vCenter Server, then select **Configure**.
4. Select **Key Providers** under **Security**.

5 Select **Add Trusted Key Providers**.

The trusted key providers that are available are shown with a status of Connected.

6 Select a trusted key provider and click **Add Key Providers**.

The trusted key provider is shown as Trusted and Connected. If this is the first trusted key provider that you add, it is marked as the default.

Note It takes a while for all the hosts to be able to get the key provider, and for the vCenter Server to update its cache. Because of the way the information is propagated, you might have to wait for a few minutes to use the key provider for key operations on some of the hosts.

Results

ESXi Trusted Hosts can now perform cryptographic operations, such as creating encrypted virtual machines.

What to do next

Encrypting a virtual machine with a trusted key provider looks the same as the virtual machine encryption user experience that was first delivered in vSphere 6.5. See Chapter 10 Use Encryption in Your vSphere Environment.

Configure the Trusted Key Provider for Trusted Hosts Using the Command Line

You can configure trusted key providers by using the command line. You can configure the default trusted key provider for the vCenter Server, or at the cluster or the folder level in the vCenter object hierarchy.

Prerequisites

- Enable the Trust Authority Administrator.
- Enable the Trust Authority State.
- Collect Information About ESXi Hosts and vCenter Server to Be Trusted.
- Import the Trusted Host Information to the Trust Authority Cluster.
- Create the Key Provider on the Trust Authority Cluster.
- Export the Trust Authority Cluster Information.
- Import the Trust Authority Cluster Information to the Trusted Hosts.

On the Trusted Cluster, you must have a role that includes the **Cryptographic operations.Manage KMS** privilege.

Procedure

1 Ensure that you are connected as an administrator to the vCenter Server of the Trusted Cluster.

   For example, you can enter `$global:defaultviservers` to show all the connected servers.

2 (Optional) If necessary, you can run the following commands to ensure that you are connected to the vCenter Server of the Trusted Cluster.

```
Disconnect-VIServer -server * -Confirm:$false
Connect-VIServer -server TrustedCluster_VC_ip_address -User admin_user -Password 'password'
```

3 Obtain the trusted key provider.

```
Get-KeyProvider
```

   You can use the `-Name` *keyprovider* option to specify a single trusted key provider.

4 Assign the `Get-KeyProvider` trusted key provider information to a variable.

   For example, this command assigns the information to the variable `$workload_kp`.

```
$workload_kp = Get-KeyProvider
```

   If you have multiple trusted key providers, you can use `Select-Object` to select one of them.

```
$workload_kp = Get-KeyProvider | Select-Object -Index 0
```

5 Register the trusted key provider.

```
Register-KeyProvider -KeyProvider $workload_kp
```

   To register additional trusted key providers, repeat Step 4 and Step 5.

Note It takes a while for all the hosts to be able to get the key provider, and for the vCenter Server to update its cache. Because of the way the information is propagated, you might have to wait for a few minutes to use the key provider for key operations on some of the hosts.

6 Set the default trusted key provider to use.

a To set the default key provider at the vCenter Server level, run the following command.

```
Set-KeyProvider -KeyProvider $workload_kp -DefaultForSystem
```

b To set the key provider at the cluster level, run the following command.

For example, this command sets the key provider for the cluster `Trusted Cluster`.

```
Add-EntityDefaultKeyProvider -KeyProvider $workload_kp -Entity 'Trusted Cluster'
```

c To set the key provider at the folder level, run the following command.

For example, this command sets the key provider for the folder `TC Folder`, which was created on the `workLoad` data center.

```
Add-EntityDefaultKeyProvider -KeyProvider $workload_kp -Entity 'TC Folder'
```

What to do next

Encrypting a virtual machine with a trusted key provider looks the same as the virtual machine encryption user experience that was first delivered in vSphere 6.5. See Chapter 10 Use Encryption in Your vSphere Environment.

Managing vSphere Trust Authority in Your vSphere Environment

After you configure vSphere Trust Authority, you can perform additional operations, such as stopping and starting services, adding hosts to clusters, and viewing the status of the Trust Authority Cluster.

You can perform tasks by using the vSphere Client, the API, and the PowerCLI cmdlets. See *vSphere Web Services SDK Programming Guide*, the *VMware PowerCLI* documentation, and the *VMware PowerCLI Cmdlets Reference* documentation.

Start, Stop, and Restart vSphere Trust Authority Services

You can start, stop, and restart vSphere Trust Authority services by using the vSphere Client.

The services that make up vSphere Trust Authority are the Attestation Service (attestd) and the Key Provider Service (kmxd).

Procedure

1 Connect to the vCenter Server of the vSphere Trust Authority Cluster by using the vSphere Client.

2 Log in as an administrator.

3 Browse to an ESXi host in the Trust Authority Cluster.

4 Select **Configure**, then select **Services** under **System**.

5 Locate the attestd service and the kmxd service.

6 Select the **Restart**, **Start**, or **Stop** operation as appropriate.

View the Trust Authority Hosts

You can view the vSphere Trust Authority hosts configured for a Trusted Cluster by using the vSphere Client.

Procedure

1 Connect to the vCenter Server of the Trusted Cluster by using the vSphere Client.

2 Log in as an administrator.

3 Select the vCenter Server instance.

4 Click the **Configure** tab and select **Trust Authority** under **Security**.

The ESXi hosts in the Trust Authority Cluster configured for the Trusted Cluster are displayed.

View the vSphere Trust Authority Cluster State

You can view the state of the vSphere Trust Authority Cluster by using the vSphere Client. The state is either enabled or disabled.

When the Trust Authority Cluster state is enabled, the Trusted Hosts in the Trusted Cluster can communicate with the Attestation Service and the Key Provider Service.

Procedure

1 Connect to the vCenter Server of the Trust Authority Cluster by using the vSphere Client.

2 Log in as an administrator.

3 Select the Trust Authority Cluster in the object hierarchy.

4 Click the **Configure** tab and select **Trust Authority Cluster** under **Trust Authority**.

The status displays as Enabled or Disabled.

Restart the Trusted Host Service

You can restart the service that runs on your Trusted Hosts.

The service, kmxa, runs on the ESXi Trusted Hosts.

Prerequisites

Access to the ESXi shell must be enabled. See Enable Access to the ESXi Shell.

Procedure

1 Use SSH or another remote console connection to start a session on the ESXi Trusted Host.

2 Log in as root.

3 Run the following command.

```
/etc/init.d/kmxa restart
```

Adding and Removing vSphere Trust Authority Hosts

You add and remove ESXi hosts to a vSphere Trust Authority Cluster by using VMware-supplied scripts.

In vSphere 7.0, you add and remove ESXi hosts to and from an existing vSphere Trust Authority Cluster or a Trusted Cluster by using VMware-supplied scripts. Starting in vSphere 7.0 Update 1, you use the remediate function to add ESXi hosts to an existing Trusted Cluster. See Add a Host to a Trusted Cluster with the vSphere Client and Add a Host to a Trusted Cluster with the CLI. In vSphere 7.0 Update 1, you still must use scripts to add ESXi hosts to an existing Trust Authority Cluster. See the VMware knowledge base articles at https://kb.vmware.com/s/article/77234 and https://kb.vmware.com/s/article/77146.

Add a Host to a Trusted Cluster with the vSphere Client

You can add ESXi hosts to an existing Trusted Cluster using the vSphere Client.

After you have initially configured a Trusted Cluster, you might want to add more ESXi hosts. However, when you add the host to a Trusted Cluster, you must take the additional step of remediation. When you remediate the Trusted Cluster, you ensure that its desired configuration state matches its applied configuration.

In the first version of vSphere Trust Authority released in vSphere 7.0, you run scripts to add a host to an existing Trusted Cluster. Starting in vSphere 7.0 Update 1, you use the remediate functionality to add a host to a Trusted Cluster. In vSphere 7.0 Update 1, you still must use scripts to add a host to an existing Trust Authority Cluster. See Adding and Removing vSphere Trust Authority Hosts.

Prerequisites

The vCenter Server for the Trusted Cluster must be running vSphere 7.0 Update 1 or later.

If you are adding an ESXi host that has a different ESXi version, or a different TPM hardware type, than what you initially configured for the Trusted Cluster, additional steps are required. You must export and import this information to the vSphere Trust Authority Cluster. See Collect Information About ESXi Hosts and vCenter Server to Be Trusted and Import the Trusted Host Information to the Trust Authority Cluster.

Required privileges: See the add hosts tasks in Required Privileges for Common Tasks.

Procedure

1 Connect to the vCenter Server of the Trusted Cluster by using the vSphere Client.

2 Log in as a Trust Authority administrator.

3 Navigate to a Trusted Cluster.

4 On the **Configure** tab, select **Configuration > Quickstart**.

5 Click **Add** in the **Add hosts** card.

6 Follow the prompts.

7 On the **Trust Authority** tab, click **Remediate**.

8 To verify that the Trusted Cluster is healthy, click **Check Health**.

Add a Host to a Trusted Cluster with the CLI

You can add ESXi hosts to an existing Trusted Cluster using the command line.

After you have initially configured a Trusted Cluster, you might want to add more ESXi hosts. However, when you add the host to a Trusted Cluster, you must take the additional step of remediation. When you remediate the Trusted Cluster, you ensure that its desired configuration state matches its applied configuration.

In the first version of vSphere Trust Authority released in vSphere 7.0, you run scripts to add a host to an existing Trusted Cluster. Starting in vSphere 7.0 Update 1, you use the remediate functionality to add a Trusted Host. In vSphere 7.0 Update 1, you still must use scripts to add a host to an existing Trust Authority Cluster. See Adding and Removing vSphere Trust Authority Hosts.

Prerequisites

- The vCenter Server for the Trusted Cluster must be running vSphere 7.0 Update 1 or later.
- PowerCLI 12.1.0 or later is required.
- Required privileges: See the add hosts tasks in Required Privileges for Common Tasks.

Procedure

1 Use whatever steps you normally do to add the ESXi host to the Trusted Cluster.

2 In a PowerCLI session, run the `Connect-VIServer` cmdlet to connect as the Trust Authority administrator to the vCenter Server of the Trusted Cluster.

```
Connect-VIServer -server TrustedCluster_VC_ip_address -User trust_admin_user -Password
'password'
```

3 To check the Trusted Cluster's status, run the `Get-TrustedClusterAppliedStatus` PowerCLI cmdlet.

```
Get-TrustedClusterAppliedStatus -TrustedCluster 'TrustedCluster'
```

4 If the Trusted Cluster is not healthy, run the `Set-TrustedCluster` cmdlet with the `-Remediate` parameter.

```
Set-TrustedCluster -TrustedCluster 'TrustedCluster' -Remediate
```

5 To verify that the Trusted Cluster is healthy, rerun the `Get-TrustedClusterAppliedStatus` cmdlet.

```
Get-TrustedClusterAppliedStatus -TrustedCluster 'TrustedCluster'
```

Decommission Trusted Hosts from a Trusted Cluster

You can remove, or decommission, Trusted Hosts from a Trusted Cluster. You can decommission one or all Trusted Hosts from a Trusted Cluster, depending on the scenario.

When you decommission a Trusted Host, the remediate function sets the desired state of the Trusted Host to that of the non-Trusted Cluster where it is moved to. The decommissioned Trusted Host becomes a regular host. The Trusted Cluster (from where the Trusted Host was moved) continues to have its desired state configuration and functions still as a Trusted Cluster.

When you remove all the Trusted Hosts from a Trusted Cluster, you decommission the Trusted Cluster. You remove both the desired state configuration and applied configuration from the Trusted Hosts and the Trusted Cluster, then move all the Trusted Hosts to a non-Trusted Cluster.

You can reuse decommissioned Trusted Hosts in your environment. For example you can reuse the hosts in a non-trusted infrastructure capacity, or as vSphere Trust Authority Hosts. You can use the decommissioned hosts in the same vCenter Server or a different vCenter Server.

For more information about Trusted Cluster configuration and health, see Trusted Cluster Health and Remediation Overview.

Prerequisites

- The vCenter Server for the Trusted Cluster must be running vSphere 7.0 Update 1 or later.
- If you use PowerCLI, version 12.1.0 or later is required.

Procedure

1 Connect to the vCenter Server of the Trusted Cluster by using the vSphere Client.

2 Log in as a Trust Authority administrator.

3 Navigate to a Trusted Cluster.

4 Decide how to decommission the Trusted Hosts from the Trusted Cluster.

Task	Steps
Keep the desired configuration state of the Trusted Cluster and the remaining Trusted Hosts	a Put hosts into Maintenance mode and move them to a new, empty cluster (that is, the cluster does not contain any hosts). b Exit Maintenance mode on the hosts. c For the new, empty cluster (not the Trusted Cluster), on the **Trust Authority** tab, click **Remediate**. Remediation removes the Trusted configuration from the moved hosts. The Trusted Cluster retains its desired state configuration.

Remove the desired configuration state and applied configuration state of all the Trusted Hosts	a In a PowerCLI session, run the `Connect-VIServer` cmdlet to connect as the Trust Authority administrator to the vCenter Server of the Trusted Cluster.
	`Connect-VIServer -server TrustedCluster_VC_ip_address -User trust_admin_user -Password 'password'`
	b Run the `Set-TrustedCluster` cmdlet, for example:
	`Set-TrustedCluster -TrustedCluster 'TrustedCluster' -State Disabled`
	The Trusted Infrastructure configuration is removed from all the Trusted Hosts, and the Trusted Cluster has its desired state configuration removed.
	c Put all hosts into Maintenance mode and move them to a different cluster.
	d Exit Maintenance mode on the hosts.

5 To verify that the Trusted Cluster is healthy, click **Check Health** on the **Trust Authority** tab for the Trusted Cluster.

What to do next

If you no longer plan to attest the specific versions of ESXi or the TPM hardware from the decommissioned ESXi hosts, update the Trust Authority Cluster's configuration, for optimal security. See the VMware knowledge base article at https://kb.vmware.com/s/article/77146.

Backing Up the vSphere Trust Authority Configuration

Use the files you exported when configuring vSphere Trust Authority as your Trust Authority backup. You can use these files to restore a Trust Authority deployment. Keep these configuration files confidential and transport them securely.

Most vSphere Trust Authority configuration and state information is stored on the ESXi hosts in the ConfigStore database. The vCenter Server Management Interface that you use to back up a vCenter Server instance does not back up the configuration information for vSphere Trust Authority. If you save and securely store the configuration files that you exported when setting up your vSphere Trust Authority environment, then you have the necessary information to restore a vSphere Trust Authority configuration. See Collect Information About ESXi Hosts and vCenter Server to Be Trusted if you must generate this information.

Change the Primary Key of a Key Provider

You can change the primary key of a key provider, for example, when you want to rotate the primary key that is used.

See Virtual Machine Encryption Best Practices for guidance about key life cycle.

Prerequisites

Create and activate a key on the key server (KMS) to be used as the new primary key for the trusted key provider. This key wraps other keys and secrets used by this trusted key provider. See your KMS vendor documentation for more information about creating keys.

Procedure

1 Run the `Set-TrustAuthorityKeyProvider` command.

   For example:

```
Set-TrustAuthorityKeyProvider -MasterKeyId Key-ID
```

2 Verify the status of the key provider.

a Assign `Get-TrustAuthorityCluster` information to a variable.

For example:

```
$vTA = Get-TrustAuthorityCluster 'vTA Cluster'
```

b Assign the `Get-TrustAuthorityKeyProvider -TrustAuthorityCluster $vTA` information to a variable.

For example:

```
$kp = Get-TrustAuthorityKeyProvider -TrustAuthorityCluster $vTA
```

c Verify the status of the key provider by running `$kp.Status`.

For example:

```
$kp.Status

KeyProviderId Health HealthDetails ServerStatus
------------- ------ ------------- ------------
domain-c8-kp4     Ok {}            {IP_address}
```

A Health status of Ok indicates that the key provider is running correctly.

Results

The new primary key is used for any new encryption operations. Data encrypted with the old primary key is still decrypted using the old key.

Trusted Host Attestation Reporting Overview

In vSphere Trust Authority, vCenter Server verifies and reports on a Trusted Host's attestation status. You can use the vSphere Client to view the attestation status of Trusted Hosts.

vSphere Trust Authority uses remote attestation for Trusted Hosts to prove the authenticity of their booted software. Attestation verifies that the Trusted Hosts are running authentic VMware software, or VMware-signed partner software. The vCenter Server of the Trusted Cluster communicates with the Trusted Host to get an internal attestation report. The attestation report specifies if the Trusted Host has attested or not with the Attestation Service running on the Trust Authority Cluster. If the Trusted Host has not attested, the attestation report also specifies an error message. The vSphere Client displays the following attestation statuses for Trusted Hosts.

Passed

The Trusted Host has attested with a vSphere Trust Authority Attestation Service, and the internal attestation report is available to vCenter Server.

Failed

The Trusted Host was not able to attest with any vSphere Trust Authority Attestation Service. The vCenter Server internal attestation report contains the error reported by the Attestation Service that the Trusted Host tried to attest with.

The vSphere Client also displays if a host was attested by vSphere Trust Authority or by vCenter Server.

When a Trusted Host is unattested, virtual machines, including encrypted virtual machines, that are running on the Trusted Host continue to be accessible. You cannot power on virtual machines on an unattested Trusted Host. However, you can still add unencrypted virtual machines. When a Trusted Host is unattested, take steps to resolve the attestation problem. For more information about attestation concepts, see vSphere Trust Authority Process Flows.

When you have configured multiple Trust Authority Hosts, there are potentially multiple attestation reports available from each host. When reporting status, the vSphere Client displays the status from the first "attested" report that it finds. If there are no "attested" reports, the vSphere Client displays the error from the first "unattested" report that it finds.

Even if you have configured multiple Trust Authority Hosts, the vSphere Client displays the status, and potentially an error message, from only one attestation report.

View the Trusted Cluster Attestation Status

You can view the attestation status of a Trusted Host by using the vSphere Client.

Prerequisites

- Both the Trusted Hosts and the vSphere Trust Authority hosts must be running ESXi 7.0 Update 1 or later.
- The vCenter Server hosts for the respective clusters must be running vSphere 7.0 Update 1 or later.

Procedure

1. Connect to the vCenter Server of the Trusted Cluster by using the vSphere Client.
2. Log in as an administrator.

   You can log in as a Trust Authority administrator or vSphere administrator.
3. Navigate to a data center and click the **Monitor** tab.
4. Click **Security**.
5. Review the Trusted Host's status in the Attestation column and read the accompanying message in the Message column.

What to do next

If there are errors, see Troubleshoot Trusted Host Attestation Problems.

Troubleshoot Trusted Host Attestation Problems

The vSphere Trust Authority attestation reporting provides a starting point for troubleshooting Trusted Host attestation errors.

Procedure

1 View the Trusted Cluster Attestation Status.

2 Use the following table to troubleshoot and resolve errors.

Error	Cause and Solution
Attestation Services not configured.	Attestation Services have not been configured. Configure the Trusted Host to use attestation services by using the Remediate action. See Remediate a Trusted Cluster.
No TPM2 device available.	Install and configure the Trusted Host to use a Trusted Platform Module (TPM). See your vendor documentation.
TPM2 endorsement public key or certificate could not be retrieved.	Check that the TPM is supported, and that it has a valid endorsement key. You might need to contact VMware Support.
Attestation report is not available.	It is possible that the Trusted Host has not finished attestation. Wait a few minutes then recheck the attestation status.
Attestation Service version is incompatible with the request.	Update the Trust Authority host running the Attestation Service to vSphere 7.0 Update 1 or later.
Attestation failed because Secure Boot is not enabled.	Check that the Trusted Host is configured to use Secure Boot. See UEFI Secure Boot for ESXi Hosts.
Attestation failed to identify the remote software version.	Import the Trusted Host's base image information to the Attestation Service. See Import the Trusted Host Information to the Trust Authority Cluster.
Attestation failed because a TPM certificate is required.	Check that the TPM is supported. Alternatively, run the following PowerCLI cmdlet to modify the com.vmware.esx.attestation.tpm2.settings to set `requireCertificateValidation` to `false`. `Set-TrustAuthorityTpm2AttestationSettings -TrustAuthorityCluster TrustedCluster -RequireCertificateValidation:$false -RequireEndorsementKey:$true`
Attestation failed due to an unknown TPM.	Import the TPM endorsement key to the Attestation Services. See Import the Trusted Host Information to the Trust Authority Cluster.
Error: vapi.send.failed.	The kmxa service might not be running on the Trusted Host or the kmxa service cannot contact the Attestation Service. Ensure that the kmxa service is started. Also, check that the Attestation Service is running. See Restart the Trusted Host Service.

Checking and Remediating Trusted Cluster Health

You can check and validate the health of a Trusted Cluster. When problems arise with the health of a Trusted Cluster, you can remediate a Trusted Cluster's configuration.

Trusted Cluster Health and Remediation Overview

If a Trusted Cluster's configuration is not healthy, you must resolve the configuration inconsistencies. You do so by remediating the Trusted Cluster. When you remediate a Trusted Cluster, you ensure that all the Trusted Hosts in the Trusted Cluster have the same trusted configuration.

A Trusted Cluster consists of a vCenter Server cluster of Trusted ESXi Hosts that are remotely attested by the Trust Authority Cluster. When you configure vSphere Trust Authority initially, you must import the Trust Authority Services information from your Trust Authority Cluster into the Trusted Cluster. The Trusted Cluster uses that configuration of components for contacting the Key Provider Service and the Attestation Service running on the Trust Authority Cluster. For more information about configuring a Trusted Cluster, see Import the Trust Authority Cluster Information to the Trusted Hosts. After you configure a Trusted Cluster, you can check and remediate its health.

Trusted Cluster Health Overview

Checking the health of a Trusted Cluster depends upon the following.

Desired state configuration

The desired state configuration is based on the Trust Authority Services information that you import into the Trusted Cluster. The desired state configuration is the Trusted Cluster's "source of truth." Think of the desired state configuration as what is initially created when you set up the Trusted Cluster.

Applied configuration

The applied configuration is the registration of the specific Attestation Services and Key Provider Services for which you have configured the Trusted Cluster. The applied configuration is what the Trusted Cluster is running currently. You can think of the applied configuration as the "run-time" configuration. The desired state configuration should match the applied configuration. However, if the applied configuration is inconsistent with the desired state configuration, the Trusted Cluster is deemed "not healthy." A Trusted Cluster that is not healthy can experience degraded performance or not function at all.

This health check is not an indicator of the overall health for either a Trusted Cluster or the vSphere Trust Authority infrastructure. The health check only compares the Trusted Cluster's desired state configuration to the applied configuration.

Trusted Cluster Remediation Overview

Remediation is the process by which vSphere Trust Authority resolves an inconsistent configuration of a Trusted Cluster. A Trusted Cluster's configuration can become inconsistent over time or due to other operational errors.

Use remediation in the following way:

- Check the Trusted Cluster health.
- If the Trusted Cluster is unhealthy, remediate it.

You can use either the vSphere Client or the CLI to check the Trusted Cluster health. See Check Trusted Cluster Health. You can also use either the vSphere Client or the CLI to remediate a Trusted Cluster. See Remediate a Trusted Cluster.

Note Remediation is also the appropriate process to use when you add a host to an existing Trusted Cluster. See Add a Host to a Trusted Cluster with the vSphere Client and Add a Host to a Trusted Cluster with the CLI.

Check Trusted Cluster Health

You can check the health status of a Trusted Cluster by using either the vSphere Client or the command line.

For more information, see Trusted Cluster Health and Remediation Overview.

Prerequisites

- The vCenter Server for the Trusted Cluster must be running vSphere 7.0 Update 1 or later.
- If you use PowerCLI, version 12.1.0 or later is required.

Procedure

1 Check the Trusted Cluster health.

Tool	Steps
vSphere Client	a Connect to the vCenter Server of the Trusted Cluster by using the vSphere Client. b Log in as a Trust Authority administrator. c Navigate to a Trusted Cluster, select **Configure**, then select **Trust Authority**. d Click **Check Health**.

CLI	a In a PowerCLI session, run the `Connect-VIServer` cmdlet to connect as the Trust Authority administrator to the vCenter Server of the Trusted Cluster.
	`Connect-VIServer -server TrustedCluster_VC_ip_address -User trust_admin_user -Password 'password'`
	b Run the `Get-TrustedClusterAppliedStatus` cmdlet, for example:
	`Get-TrustedClusterAppliedStatus -TrustedCluster 'TrustedCluster'`

2 If there are errors, see Remediate a Trusted Cluster.

Remediate a Trusted Cluster

You can remediate the configuration of a Trusted Cluster by using either the vSphere Client or the command line.

Prerequisites

The vCenter Server for the Trusted Cluster must be running vSphere 7.0 Update 1 or later.

Procedure

1 Connect to the vCenter Server of the Trusted Cluster.

Tool	Steps
vSphere Client	a Connect to the vCenter Server of the Trusted Cluster by using the vSphere Client. b Log in as a Trust Authority administrator.
CLI	In a PowerCLI session, run the `Connect-VIServer` cmdlet to connect as the Trust Authority administrator to the vCenter Server of the Trusted Cluster. `Connect-VIServer -server TrustedCluster_VC_ip_address -User trust_admin_user -Password 'password'`

2 Remediate the Trusted Cluster then recheck the Trusted Cluster health.

Tool	Steps
vSphere Client	a Navigate to a Trusted Cluster. b Select **Configure**, then select **Trust Authority**. c Click **Remediate**. d Click **Check Health**.

CLI	a Run the `Set-TrustedCluster` cmdlet with the `-Remediate` parameter, for example:
	`Set-TrustedCluster -TrustedCluster 'TrustedCluster' -Remediate`
	b Run the `Get-TrustedClusterAppliedStatus` cmdlet, for example:
	`Get-TrustedClusterAppliedStatus -TrustedCluster 'TrustedCluster'`

Use Encryption in Your vSphere Environment

10

Whether you use a standard key provider, trusted key provider, or vSphere Native Key Provider, using encryption in your vSphere environment requires some preparation.

After your environment is set up, you can use the vSphere Client to create encrypted virtual machines and virtual disks and encrypt existing virtual machines and disks.

You can perform additional tasks by using the API and by using the `crypto-util` CLI. See the *vSphere Web Services SDK Programming Guide* for the API documentation and the `crypto-util` command-line help for details about that tool.

Create an Encryption Storage Policy

Before you can create encrypted virtual machines, you must create an encryption storage policy. You create the storage policy once, and assign it each time you encrypt a virtual machine or virtual disk.

If you want to use virtual machine encryption with other I/O filters, or to use the **Create VM Storage Policy** wizard in the vSphere Client, see the *vSphere Storage* documentation for details.

Prerequisites

- Set up the connection to a key provider.

  Although you can create a VM Encryption storage policy without the key provider connection in place, you cannot perform encryption tasks until trusted connection with the key provider is established.
- Required privileges: **Cryptographic operations.Manage encryption policies**.

Procedure

1. Log in to the vCenter Server by using the vSphere Client.
2. Select **Home**, click **Policies and Profiles**, then click **VM Storage Policies**.
3. Click **Create**.
4. Select the vCenter Server, enter a policy name, optionally enter a description, then click **Next**.
5. On the **Policy structure** page, check **Enable host based roles** then click **Next**.

6 On the **Host based services** page, select **Use storage policy component**, choose **Default encryption properties** from the drop-down menu, then click **Next**.

7 On the **Storage compatibility** page, leave **Compatible** selected, select a datastore, then click **Next**.

8 Review the information and click **Finish**.

Results

The VM Encryption storage policy is added to the list, and is available for use when encrypting a virtual machine.

Enable Host Encryption Mode Explicitly

Host encryption mode must be enabled if you want to perform encryption tasks, such as creating an encrypted virtual machine, on an ESXi host. In most cases, host encryption mode is enabled automatically when you perform an encryption task.

Sometimes, turning on encryption mode explicitly is necessary. See Prerequisites and Required Privileges for Encryption Tasks.

Prerequisites

Required privilege: **Cryptographic operations.Register host**

Procedure

1 Log in to the vCenter Server by using the vSphere Client.

2 Browse to the ESXi host and click **Configure**.

3 Under System, click **Security Profile**.

4 Click **Edit** in the Host Encryption Mode panel.

5 Select **Enabled** and click **OK**.

Disable Host Encryption Mode Using the API

Host encryption mode is enabled automatically when you perform an encryption task, if the user has sufficient privilege to enable the encryption mode. After host encryption mode is enabled, all core dumps are encrypted to avoid the release of sensitive information to support personnel. If you no longer use virtual machine encryption with an ESXi host, you can disable encryption mode.

After encryption mode is enabled for an ESXi host, you might need to disable it. For example, you might need to disable encryption mode to generate an ESXi support bundle (using the `vm-support` command). Using the Disable Host Encryption mode toggle (**Host > Configure > Security Profile > Edit Host Encryption Mode**) does not work when key material exists on the host.

You can use the API to disable host encryption mode by invoking the `CryptoManagerHostDisable` API method.

The crypto modes, or states, defined for an ESXi host are:

- pendingIncapable: The host is crypto disabled, that is, the host cannot perform vSphere Virtual Machine Encryption operations.
- incapable: The host is not safe for receiving sensitive material.
- prepared: The host is prepared for receiving sensitive material but does not have a host key set yet.
- safe: The host is crypto safe (enabled), and has a host key set, that is, vSphere Virtual Machine Encryption operations are possible.

After you invoke `CryptoManagerHostDisable` on a host, the crypto state of the host changes as follows:

- If the original host crypto state is incapable or prepared, the host crypto state is changed to incapable.
- If the original host crypto state is safe, the host crypto state is changed to pendingIncapable.
- If the host crypto state is pendingIncapable, the host crypto state is still pendingIncapable.

This task shows how to disable host encryption mode by using the vCenter Server Managed Object Browser (MOB). For more information about using the API, see the *vSphere Web Services API* documentation at https://developer.vmware.com/apis/968/vsphere.

Procedure

1. Log in to the vCenter Server as an administrator.
2. Unregister all encrypted virtual machines from the ESXi host whose encryption mode you want to disable.
3. Access the MOB on the vCenter Server.

   ```
   https://vcenter_server/mob
   ```

4. Invoke the `CryptoManagerHostDisable` method on a host.
   a. Under content name, click **content**.
   b. Under rootFolder, click **group-D1 (Datacenters)**.
   c. Under childEntity, click the appropriate datacenter.
   d. Under hostFolder, click the appropriate host.
   e. Under childEntity, click the appropriate cluster.
   f. Under host, click the appropriate host.
   g. Under configManager, click **configManager**.

h Under cryptoManager, click **CryptoManagerHost-***number*.

i Click **CryptoManagerHostDisable**.

The host crypto state is changed to either pendingIncapable or incapable, depending on its original crypto state.

5 Repeat step 4 for other hosts on which you want to disable encryption mode.

6 Reboot the hosts.

Results

Once the host encryption mode is disabled, you cannot perform encryption operations, such as adding encrypted virtual machines, unless you re-enable the host encryption mode.

Note After you reboot an ESXi host on which you disabled encryption mode, if the host crypto state was originally pendingIncapable, the host crypto state is still pendingIncapable. To re-enable host encryption mode, re-access the vCenter Server MOB and invoke the `ConfigureCryptoKey` API method. When re-enabling host encryption mode, use the original host key ID if the host crypto state is pendingIncapable.

Create an Encrypted Virtual Machine

After you set up the KMS, you can create encrypted virtual machines.

This task describes how to create an encrypted virtual machine using the vSphere Client. The vSphere Client filters by virtual machine encryption storage policies, easing creation of encrypted virtual machines.

Note Creating an encrypted virtual machine is faster and uses fewer storage resources than encrypting an existing virtual machine. If possible, encrypt virtual machines during the creation process.

Prerequisites

- Establish a trusted connection with the KMS and select a default KMS.
- Create an encryption storage policy, or use the bundled sample, VM Encryption Policy.
- Ensure that the virtual machine is powered off.
- Verify that you have the required privileges:
  - **Cryptographic operations.Encrypt new**
  - If the host encryption mode is not Enabled, you also need **Cryptographic operations.Register host**.

Procedure

1 Connect to vCenter Server by using the vSphere Client.

2 Select an object in the inventory that is a valid parent object of a virtual machine, for example, an ESXi host or a cluster.

3 Right-click the object and select **New Virtual Machine**.

4 Follow the prompts to create an encrypted virtual machine.

Option	Action
Select a creation type	Create a new virtual machine.
Select a name and folder	Specify a unique name and target location for the virtual machine.
Select a compute resource	Specify an object for which you have privileges to create encrypted virtual machines. See Prerequisites and Required Privileges for Encryption Tasks.
Select storage	Select the **Encrypt this virtual machine** check box. Virtual machine storage policies that include encryption appear. Select a virtual machine storage policy (the bundled sample is VM Encryption Policy), and select a compatible datastore.
Select compatibility	Select the compatibility. You can migrate an encrypted virtual machine only to hosts with compatibility ESXi 6.5 and later.
Select a guest OS	Select a guest OS that you plan to install on the virtual machine later.
Customize hardware	Customize the hardware, for example, by changing disk size or CPU. (Optional) Select the **VM Options** tab, and expand **Encryption**. Select which disks to exclude from encryption. When you deselect a disk, only the VM Home and any other selected disks are encrypted. Any New Hard disk that you add is encrypted. You can change the storage policy for individual hard disks later.
Ready to complete	Review the information and click **Finish**.

Clone an Encrypted Virtual Machine

When you clone an encrypted virtual machine, the clone is encrypted with the same keys. To change keys for the clone, perform a recrypt of the clone using the API. See *vSphere Web Services SDK Programming Guide*.

You can perform the following operations during clone.

- Create an encrypted virtual machine from an unencrypted virtual machine or template virtual machine.
- Create a decrypted virtual machine from an encrypted virtual machine or template virtual machine.
- Recrypt the destination virtual machine with different keys from that of source virtual machine.

You can create an instant clone virtual machine from an encrypted virtual machine with the caveat that the instant clone shares the same key with the source virtual machine. You cannot recrypt keys on either the source or the instant clone virtual machine. See *vSphere Web Services SDK Programming Guide*.

Prerequisites

- Establish a trusted connection with the KMS and select a default KMS.
- Create an encryption storage policy, or use the bundled sample, VM Encryption Policy.
- Required privileges:
  - **Cryptographic operations.Clone**
  - **Cryptographic operations.Encrypt**
  - **Cryptographic operations.Decrypt**
  - **Cryptographic operations.Recrypt**
  - If the host encryption mode is not Enabled, you also must have **Cryptographic operations.Register host** privileges.

Procedure

1. Browse to the virtual machine in the vSphere Client inventory.
2. To create a clone of an encrypted machine, right-click the virtual machine, select **Clone** > **Clone to Virtual Machine**, and follow the prompts.

Option	Action
Select a name and folder	Specify a name and target location for the clone.
Select a compute resource	Specify an object for which you have privileges to create encrypted virtual machines. See Prerequisites and Required Privileges for Encryption Tasks.
Select storage	Make a selection in the **Select virtual disk format** menu and select a datastore. You can change the storage policy as part of the clone operation. For example, changing from using an encryption policy to a non-encryption policy decrypts the disks.
Select clone options	Select clone options, as discussed in the *vSphere Virtual Machine Administration* documentation.
Ready to complete	Review the information and click **Finish**.

3 (Optional) Change the keys for the cloned virtual machine.

By default, the cloned virtual machine is created with the same keys as its parent. Best practice is to change the cloned virtual machine's keys to ensure that multiple virtual machines do not have the same keys.

a Decide upon a shallow or deep recrypt.

To use a different DEK and KEK, perform a deep recrypt of the cloned virtual machine. To use a different KEK, perform a shallow recrypt of the cloned virtual machine. For a deep recrypt, you must power off the virtual machine. You can perform a shallow recrypt operation while the virtual machine is powered on, and if the virtual machine has snapshots present. Shallow recrypt of an encrypted virtual machine with snapshots is permitted only on a single snapshot branch (disk chain). Multiple snapshot branches are not supported. If the shallow recrypt fails before updating all links in the chain with the new KEK, you can still access the encrypted virtual machine if you have the old and new KEKs.

b Perform a recrypt of the clone using the API. See *vSphere Web Services SDK Programming Guide*.

Encrypt an Existing Virtual Machine or Virtual Disk

You can encrypt an existing virtual machine or virtual disk by changing its storage policy. You can encrypt virtual disks only for encrypted virtual machines.

This task describes how to encrypt an existing virtual machine or virtual disk using the vSphere Client.

 (Encrypting Virtual Machines with the vSphere Client)

Prerequisites

- Establish a trusted connection with the KMS and select a default KMS.
- Create an encryption storage policy, or use the bundled sample, VM Encryption Policy.
- Ensure that the virtual machine is powered off.
- Verify that you have the required privileges:
  - **Cryptographic operations.Encrypt new**
  - If the host encryption mode is not Enabled, you also need **Cryptographic operations.Register host**.

Procedure

1 Connect to vCenter Server by using the vSphere Client.

2 Right-click the virtual machine that you want to change and select **VM Policies > Edit VM Storage Policies**.

   You can set the storage policy for the virtual machine files, represented by VM home, and the storage policy for virtual disks.

3 Select the storage policy.

   - To encrypt the VM and its hard disks, select an encryption storage policy and click **OK**.
   - To encrypt the VM but not the virtual disks, toggle on **Configure per disk**, select the encryption storage policy for VM Home and other storage policies for the virtual disks, and click **OK**.

   Prior to vSphere 7.0 Update 3i, you cannot encrypt the virtual disk of an unencrypted virtual machine. In vSphere Update 3i and later, the vSphere Client prompts if you want to reconfigure the unencrypted virtual machine with the encrypted disk by encrypting VM Home.

4 If you prefer, you can encrypt the virtual machine, or both virtual machine and disks, from the **Edit Settings** menu in the vSphere Client.

   a Right-click the virtual machine and select **Edit Settings**.

   b Select the **VM Options** tab, and open **Encryption**. Choose an encryption policy. If you deselect all disks, only the VM home is encrypted.

   c Click **OK**.

Decrypt an Encrypted Virtual Machine or Virtual Disk

You can decrypt a virtual machine, its disks, or both, by changing the storage policy.

This task describes how to decrypt an encrypted virtual machine using the vSphere Client.

All encrypted virtual machines require encrypted vMotion. During virtual machine decryption, the Encrypted vMotion setting remains. To change this setting so that Encrypted vMotion is no longer used, change the setting explicitly.

This task explains how to perform decryption using storage policies. For virtual disks, you can also perform decryption using the **Edit Settings** menu.

Note In the Virtual Machine Details pane, a vTPM-enabled virtual machine displays both a lock icon and an "Encrypted with *key_provider*" message. To remove a vTPM from a virtual machine, see Remove Virtual Trusted Platform Module from a Virtual Machine.

Prerequisites

- The virtual machine must be encrypted.
- The virtual machine must be powered off or in maintenance mode.
- Required privileges: **Cryptographic operations.Decrypt**

Procedure

1 Connect to vCenter Server by using the vSphere Client.

2 Right-click the virtual machine that you want to change and select **VM Policies > Edit VM Storage Policies**.

   You can set the storage policy for the virtual machine files, represented by VM home, and the storage policy for virtual disks.

3 Select a storage policy.

   - To decrypt the VM and its hard disks, toggle off **Configure per disk**, select a storage policy from the drop-down menu, and click **OK**.
   - To decrypt a virtual disk but not the virtual machine, toggle on **Configure per disk**, select the encryption storage policy for VM Home and other storage policies for the virtual disks, and click **OK**.

   You cannot decrypt the virtual machine and leave the disk encrypted.

4 If you prefer, you can use the vSphere Client to decrypt the virtual machine and disks from the **Edit Settings** menu.

   a Right-click the virtual machine and select **Edit Settings**.

   b Select the **VM Options** tab and expand **Encryption**.

   c To decrypt the VM and its hard disks, choose **None** from the **Encrypt VM** drop-down menu.

   d To decrypt a virtual disk but not the virtual machine, deselect the disk.

   e Click **OK**.

5 (Optional) You can change the Encrypted vMotion setting.

   a Right-click the virtual machine and click **Edit Settings**.

   b Click **VM Options**, and open **Encryption**.

   c Set the **Encrypted vMotion** value.

Change the Encryption Policy for Virtual Disks

When you create an encrypted virtual machine from the vSphere Client, you can select which virtual disks that you add during virtual machine creation are encrypted. You can decrypt virtual disks that are encrypted by using the **Edit VM Storage Policies** option.

Note An encrypted virtual machine can have virtual disks that are not encrypted. However, an unencrypted virtual machine cannot have encrypted virtual disks.

See Virtual Disk Encryption.

This task describes how to change the encryption policy using storage policies. You can also use the **Edit Settings** menu to make this change.

Prerequisites

- You must have the **Cryptographic operations.Manage encryption policies** privilege.
- Ensure that the virtual machine is powered off.

Procedure

1. Connect to vCenter Server by using the vSphere Client.
2. Right-click the virtual machine and select **VM Policies > Edit VM Storage Policies** .
3. Change the storage policy.
   - To change the storage policy for the VM and its hard disks, select an encryption storage policy and click **OK**.
   - To encrypt the VM but not the virtual disks, toggle on **Configure per disk**, select the encryption storage policy for VM Home and other storage policies for the virtual disks, and click **OK**.

   You cannot encrypt the virtual disk of an unencrypted VM.
4. If you prefer, you can change the storage policy from the **Edit Settings** menu.
   a. Right-click the virtual machine and select **Edit Settings**.
   b. Select the **Virtual Hardware** tab, expand a hard disk, and select an encryption policy from the drop-down menu.
   c. Click **OK**.

Resolve Missing Key Issues

If the ESXi host cannot get the key (KEK) from vCenter Server for an encrypted virtual machine or an encrypted virtual disk, the encrypted VM becomes locked. After you make the keys available on the KMS, you can unlock a locked encrypted virtual machine.

Under certain circumstances when using a standard key provider, the ESXi host cannot get the key encryption key (KEK) for an encrypted virtual machine or an encrypted virtual disk from vCenter Server. In that case, you can still unregister or reload the virtual machine. However, you cannot perform other virtual machine operations such as powering on the virtual machine. After taking the necessary steps to make the required keys available on the KMS, you can unlock a locked encrypted virtual machine by using the vSphere Client.

If the virtual machine key is not available, a vCenter Server alarm notifies you and the state of the virtual machine displays as invalid. The virtual machine cannot power on. If the virtual machine key is available, but a key for an encrypted disk is not available, the virtual machine state does not display as invalid. However, the virtual machine cannot power on and the following error results:

```
The disk [/path/to/the/disk.vmdk] is encrypted and a required key was not found.
```

Note The following procedure illustrates the situations that can cause a virtual machine to become locked, the corresponding alarms and event logs that appear, and what to do in each case.

Procedure

1 If the problem is the connection between the vCenter Server system and the KMS, vCenter Server generates a virtual machine alarm. Also, an error message appears in the event log.

   Restore the connection to the KMS. When the KMS and keys become available, unlock the locked virtual machines. See Unlock Locked Virtual Machines. You can also reboot the host and re-register the virtual machine to unlock it after restoring the connection.

   Losing the connection to the KMS does not automatically lock the virtual machine. The virtual machine only enters a locked state if the following conditions are met:

   - The key is not available on the ESXi host.
   - vCenter Server cannot retrieve keys from the KMS.

   After each reboot, the ESXi host must be able to reach vCenter Server. vCenter Server requests the key with the corresponding ID from the KMS and makes it available to ESXi.

   Note In vSphere 7.0 Update 2 and later, you can persist encryption keys across ESXi reboots. See Key Persistence Overview.

   If, after restoring connection to the key provider, the virtual machine remains locked, see Unlock Locked Virtual Machines.

2 If the connection is restored, register the virtual machine. If an error results, or if the operation succeeds but the virtual machine is in a locked state, verify that you have the **Cryptographic operations.RegisterVM** privilege for the vCenter Server system.

   This privilege is not required for powering on an encrypted virtual machine if the key is available. This privilege is required for registering the virtual machine if the key has to be retrieved.

3 If the key is no longer available on the KMS, vCenter Server generates a virtual machine alarm. Also, an error message appears in the event log.

   Ask the KMS administrator to restore the key. You might encounter an inactive key if you are powering on a virtual machine that had been removed from the inventory and that had not been registered for a long time. It also happens if you reboot the ESXi host, and the KMS is not available.

   a Retrieve the key ID by using the Managed Object Browser (MOB) or the vSphere API.

     Retrieve the `keyId` from `VirtualMachine.config.keyId.keyId`.

   b Ask the KMS administrator to reactivate the key that is associated with that key ID.

   c After restoring the key, see Unlock Locked Virtual Machines.

   If the key can be restored on the KMS, vCenter Server retrieves it and pushes it to the ESXi host the next time it is needed.

4 If the KMS is accessible and the ESXi host is powered on, but the vCenter Server system is unavailable, follow these steps to unlock virtual machines.

   a Restore the vCenter Server system, or set up a different vCenter Server system, then establish trust with the KMS.

     You must use the same key provider name, but the KMS IP address can be different.

   b Reregister all virtual machines that are locked.

     The new vCenter Server instance retrieves the keys from the KMS and the virtual machines are unlocked.

5 If the keys are missing only on the ESXi host, vCenter Server generates a virtual machine alarm and the following message appears in the event log:

   ```
   Virtual machine is locked because keys are missing on host.
   ```

   The vCenter Server system can retrieve the missing keys from the key provider. No manual recovery of keys is required. See Unlock Locked Virtual Machines.

Unlock Locked Virtual Machines

A vCenter Server alarm notifies you when an encrypted virtual machine is in a locked state. You can unlock a locked encrypted virtual machine by using the vSphere Client (HTML5-based client) after taking the necessary steps to make the required keys available on the KMS.

Prerequisites

- Verify that you have the required privileges: **Cryptographic operations.RegisterVM**
- Other privileges might be required for optional tasks such as enabling host encryption.
- Before unlocking a locked virtual machine, troubleshoot the cause of the lock and attempt to fix the problem manually. See Resolve Missing Key Issues.

Procedure

1 Connect to vCenter Server by using the vSphere Client.

2 Navigate to the virtual machine's **Summary** tab.

When a virtual machine is locked, the Virtual Machine Locked alarm appears.

3 Decide if you want to either acknowledge the alarm, or reset the alarm to green but not unlock the virtual machine now.

When you click either **Acknowledge** or **Reset to green**, the alarm goes away, but the virtual machine remains locked until you unlock it.

4 Navigate to the **Monitor** tab of the virtual machine and click **Events**.

The **Events** pane displays information about why the virtual machine is locked.

5 Perform suggested troubleshooting before you unlock the virtual machine.

6 Navigate to the **Summary** tab of the virtual machine.

A Virtual Machine Locked Alarm appears.

7 Select **Unlock VM** from the **Actions** drop-down menu on the right side.

Resolve ESXi Host Encryption Mode Issues

Under certain circumstances, the ESXi host's encryption mode can become disabled.

An ESXi host requires that host encryption mode is enabled if it contains any encrypted virtual machines. If the host detects it is missing its host key, or if the key provider is unavailable, the host might fail to enable the encryption mode. vCenter Server generates an alarm when the host encryption mode cannot be enabled.

Procedure

1 If the problem is the connection between the vCenter Server system and the key provider, an alarm is generated and an error message appears in the event log.

You must restore the connection to the key provider that contains the encryption keys in question.

2 If keys are missing, an alarm is generated and an error message appears in the event log.

You must ensure that the keys are present in the key provider. Consult the documentation for your key management vendor for information about restoring from backup.

What to do next

If, after restoring connection to the key provider, or manually recovering keys to the key provider, the host's encryption mode remains disabled, re-enable the host encryption mode. See Re-Enable ESXi Host Encryption Mode.

Re-Enable ESXi Host Encryption Mode

Starting with vSphere 6.7, a vCenter Server alarm notifies you when an ESXi host's encryption mode has become disabled. You can re-enable the host encryption mode if it has become disabled.

Prerequisites

- Verify that you have the required privileges: **Cryptographic operations.Register host**
- Before re-enabling encryption mode, troubleshoot the cause and attempt to fix the problem manually.

Procedure

1. Connect to vCenter Server by using the vSphere Client.
2. Navigate to the ESXi host's **Summary** tab.

   When the encryption mode is disabled, the Host Requires Encryption Mode Enabled alarm appears.
3. Decide if you want to either acknowledge the alarm, or reset the alarm to green but not re-enable the host encryption mode now.

   When you click either **Acknowledge** or **Reset to green**, the alarm goes way, but the host's encryption mode remains disabled until you re-enable it.
4. Navigate to the ESXi host's **Monitor** tab and click **Events** to get more information on why encryption mode is disabled.

   Perform suggested troubleshooting before you re-enable the encryption mode.
5. On the **Summary** tab, click **Enable Host Encryption Mode** to re-enable host encryption.

   A message appears, warning that encryption key data is transmitted to the host.
6. Click **Yes**.

Set Key Management Server Certificate Expiration Threshold

By default, vCenter Server notifies you 30 days before your Key Management Server (KMS) certificates expire. You can change this default value.

KMS certificates have an expiration date. When the threshold for the expiration date is reached, an alarm notifies you.

vCenter Server and key providers exchange two types of certificates: server and client. The VMware Endpoint Certificate Store (VECS) on the vCenter Server system stores the server certificates and one client certificate per key provider. Because there are two certificate types, there are two alarms for each certificate type (one for client, one for server).

Procedure

1 Log in to a vCenter Server system by using the vSphere Client.

2 Select the vCenter Server system in the object hierarchy.

3 Click **Configure**.

4 Under **Settings**, click **Advanced Settings**, and click **Edit Settings**.

5 Click the **Filter** icon and enter `vpxd.kmscert.threshold`, or scroll to the configuration parameter itself.

6 Enter your value in days and click **Save**.

vSphere Virtual Machine Encryption and Core Dumps

If your environment uses vSphere Virtual Machine Encryption, and if an error occurs on the ESXi host, the resulting core dump is encrypted to protect customer data. Core dumps that are included in the vm-support package are also encrypted.

Note Core dumps can contain sensitive information. Follow your organization's data security and privacy policy when handling core dumps.

Core Dumps on ESXi Hosts

When an ESXi host, a user world, or a virtual machine fails, a core dump is generated, and the host reboots. If the ESXi host has encryption mode enabled, the core dump is encrypted using a key that is in the ESXi key cache. This key comes from the KMS. See How vSphere Virtual Machine Encryption Protects Your Environment for background information.

When an ESXi host is cryptographically "safe," and a core dump is generated, an event is created. The event indicates that a core dump occurred along with the following information: world name, occurring times, keyID of the key used to encrypt the core dump, and core dump filename. You can view the event in the Events viewer under **Tasks and Events** for the vCenter Server.

The following table shows encryption keys used for each core dump type, by vSphere release.

Table 10-1. Core Dump Encryption Keys

Core Dump Type	Encryption Key (ESXi 6.5)	Encryption Key (ESXi 6.7 and Later)
ESXi Kernel	Host Key	Host Key
User World (hostd)	Host Key	Host Key
Encrypted Virtual Machine (VM)	Host Key	Virtual Machine Key

What you can do after an ESXi host reboot depends on several factors.

- In most cases, vCenter Server retrieves the key for the host from the KMS and attempts to push the key to the ESXi host after reboot. If the operation is successful, you can generate the vm-support package and you can decrypt or re-encrypt the core dump. See Decrypt or Re-Encrypt an Encrypted Core Dump.
- If vCenter Server cannot connect to the ESXi host, you might be able to retrieve the key from the KMS. See Resolve Missing Key Issues.
- If the host used a custom key, and that key differs from the key that vCenter Server pushes to the host, you cannot manipulate the core dump. Avoid using custom keys.

Core Dumps and vm-support Packages

When you contact VMware Technical Support because of a serious error, your support representative usually asks you to generate a vm-support package. The package includes log files and other information, including core dumps. If your support representatives cannot resolve the issues by looking at log files and other information, they might ask you to decrypt the core dumps and make relevant information available. To protect sensitive information such as keys, follow your organization's security and privacy policy. See Collect a vm-support Package for an ESXi Host That Uses Encryption.

Core Dumps on vCenter Server Systems

A core dump on a vCenter Server system is not encrypted. vCenter Server already contains potentially sensitive information. At the minimum, ensure that the vCenter Server is protected. See Chapter 4 Securing vCenter Server Systems. You might also consider turning off core dumps for the vCenter Server system. Other information in log files can help determine the problem.

Collect a vm-support Package for an ESXi Host That Uses Encryption

If host encryption mode is enabled for the ESXi host, any core dumps in the `vm-support` package are encrypted. You can collect the package from the vSphere Client, and you can specify a password if you expect to decrypt the core dump later.

The `vm-support` package includes log files, core dump files, and more.

Prerequisites

Inform your support representative that host encryption mode is enabled for the ESXi host. Your support representative might ask you to decrypt core dumps and extract relevant information.

Note Core dumps can contain sensitive information. Follow your organization's security and privacy policy to protect sensitive information such as host keys.

Procedure

1. Log in to the vCenter Server system with the vSphere Client.
2. Click **Hosts & Clusters**, and right-click the ESXi host.

3 Select **Export System Logs**.

4 In the dialog box, select **Password for encrypted core dumps**, and specify and confirm a password.

5 Leave the defaults for other options or make changes if requested by VMware Technical Support, and click **Export Logs**.

If you have not configured your browser to ask where to save files before downloading, the download starts. If you have configured your browser to ask where to save files, specify a location for the file.

6 If your support representative asked you to decrypt the core dump in the `vm-support` package, log in to any ESXi host and follow these steps.

a Log in to the ESXi host and connect to the directory where the `vm-support` package is located.

The filename follows the pattern esx.date_and_time.tgz.

b Make sure that the directory has enough space for the package, the uncompressed package, and the recompressed package, or move the package.

c Extract the package to the local directory.

```
vm-support -x *.tgz .
```

The resulting file hierarchy might contain core dump files for the ESXi host, usually in `/var/core`, and might contain multiple core dump files for virtual machines.

d Decrypt each encrypted core dump file separately.

```
crypto-util envelope extract --offset 4096 --keyfile vm-support-incident-key-file
--password encryptedZdump decryptedZdump
```

vm-support-incident-key-file is the incident key file that you find at the top level in the directory.

encryptedZdump is the name of the encrypted core dump file.

decryptedZdump is the name for the file that the command generates. Make the name similar to the *encryptedZdump* name.

e Provide the password that you specified when you created the `vm-support` package.

f Remove the encrypted core dumps, and compress the package again.

```
vm-support --reconstruct
```

7 Remove any files that contain confidential information.

Decrypt or Re-Encrypt an Encrypted Core Dump

You can decrypt or re-encrypt an encrypted core dump on your ESXi host by using the `crypto-util` CLI.

You can decrypt and examine the core dumps in the vm-support package yourself. Core dumps might contain sensitive information. Follow your organization's security and privacy policy to protect sensitive information such as keys.

For details about re-encrypting a core dump and other features of `crypto-util`, see the command-line help.

Note `crypto-util` is for advanced users.

Prerequisites

The key that was used to encrypt the core dump must be available on the ESXi host that generated the core dump.

Procedure

1 Log directly in to the ESXi host on which the core dump happened.

If the ESXi host is in lockdown mode, or if SSH access is disabled, you might have to enable access first.

2 Determine whether the core dump is encrypted.

Option	Description
Monitor core dump	`crypto-util envelope describe vmmcores.ve`
zdump file	`crypto-util envelope describe --offset 4096 zdumpFile`

3 Decrypt the core dump, depending on its type.

Option	Description
Monitor core dump	`crypto-util envelope extract vmmcores.ve vmmcores`
zdump file	`crypto-util envelope extract --offset 4096 zdumpEncrypted zdumpUnencrypted`

Enable and Disable Key Persistence on an ESXi Host

You must enable key persistence on an ESXi host. It is not enabled by default.

For conceptual information about key persistence, see Key Persistence Overview.

Prerequisites

Requirements to enable key persistence:

- ESXi 7.0 Update 2 or later
- ESXi host installed with TPM 2.0
- Have access to the ESXCLI command set. You can run ESXCLI commands remotely, or run them in the ESXi Shell.

Note Key persistence is not necessary when using vSphere Native Key Provider. vSphere Native Key Provider is designed out-of-the-box to run without requiring access to a key server.

For additional security, the TPM can also use a sealing policy to prevent tampering during ESXi host boot. See TPM Sealing Policies Overview.

Procedure

1. Start a session on the ESXi host by using SSH or another remote console connection.
2. Log in as root.
3. Verify that the ESXi host is in TPM mode.

   ```
   esxcli system settings encryption get
   ```

   If the Mode appears as NONE, you must enable the TPM in the firmware of the host, and set the mode by running the following command.

   ```
   esxcli system settings encryption set --mode=TPM
   ```

4. Enable or disable key persistence.
   a. To enable key persistence:

      ```
      esxcli system security keypersistence enable
      ```

   b. To disable persistence:

      ```
      esxcli system security keypersistence disable --remove-all-stored-keys
      ```

Rekey an Encrypted Virtual Machine Using the vSphere Client

You can use the vSphere Client to perform a shallow rekey of an encrypted virtual machine. You might perform a rekey of an encrypted virtual machine for business or compliance reasons.

A shallow rekey, or rekey (also called a shallow recrypt), enables you to use a new (and different) Key Encryption Key (KEK) on an encrypted virtual machine. You can perform a rekey operation while the virtual machine is powered on. You can also perform a rekey if the virtual machine has snapshots present. Rekeying of an encrypted virtual machine with snapshots is permitted only on a single snapshot branch (disk chain). Multiple snapshot branches are not supported. If the rekey fails before updating all links in the chain with the new KEK, you can still access the encrypted virtual machine if you have the old and new KEKs.

Prerequisites

Required privilege: **Cryptographic operations.Manage key servers**

Procedure

1. Log in to the vCenter Server system with the vSphere Client.
2. Browse the inventory list and select the encrypted virtual machine.
3. Right-click the encrypted virtual machine and select **VM Policies**.
4. Select **Re-encrypt**.
5. Click **Yes**.

   The encrypted virtual machine is rekeyed with the new KEK.

   Note If the rekey fails, the events subsystem posts the following event:

   ```
   com.vmware.vc.vm.crypto.RekeyFail
   ```

Securing Virtual Machines with Virtual Trusted Platform Module

11

With the Virtual Trusted Platform Module (vTPM) feature, you can add a TPM 2.0 virtual cryptoprocessor to a virtual machine.

A vTPM is a software-based representation of a physical Trusted Platform Module 2.0 chip. A vTPM acts as any other virtual device. You can add a vTPM to a virtual machine in the same way you add virtual CPUs, memory, disk controllers, or network controllers. A vTPM does not require a hardware Trusted Platform Module chip.

Read the following topics next:

Virtual Trusted Platform Module Overview

A virtual Trusted Platform Module (vTPM) is a software-based representation of a physical Trusted Platform Module 2.0 chip. A vTPM acts as any other virtual device.

What Is a vTPM

vTPMs provide hardware-based, security-related functions such as random number generation, attestation, key generation, and more. When added to a virtual machine, a vTPM enables the guest operating system to create and store keys that are private. These keys are not exposed to the guest operating system itself. Therefore, the virtual machine attack surface is reduced. Usually, compromising the guest operating system compromises its secrets, but enabling a vTPM greatly reduces this risk. These keys can be used only by the guest operating system for encryption or signing. With an attached vTPM, a client can remotely attest the identity of the virtual machine, and verify the software that it is running.

You can add a vTPM to either a new or an existing virtual machine. A vTPM depends on virtual machine encryption to secure vital TPM data. When you configure a vTPM, the virtual machine files are encrypted but not the disks. You can choose to add encryption explicitly for the virtual machine and its disks.

When you back up a virtual machine enabled with a vTPM, the backup must include all virtual machine data, including the `*.nvram` file. If your backup does not include the `*.nvram` file, you cannot restore a virtual machine with a vTPM. Also, because the VM home files of a vTPM-enabled virtual machine are encrypted, ensure that the encryption keys are available at the time of a restore.

A vTPM does not require a physical Trusted Platform Module (TPM) 2.0 chip to be present on the ESXi host. However, if you want to perform host attestation, an external entity, such as a TPM 2.0 physical chip, is required. See Securing ESXi Hosts with Trusted Platform Module.

Note By default, no storage policy is associated with a virtual machine that has been enabled with a vTPM. Only the virtual machine files (VM Home) are encrypted. If you prefer, you can choose to add encryption explicitly for the virtual machine and its disks, but the virtual machine files would have already been encrypted.

vSphere Requirements for vTPMs

To use a vTPM, your vSphere environment must meet these requirements:

- Virtual machine requirements:
  - EFI firmware
  - Hardware version 14 and later
- Component requirements:
  - vCenter Server 6.7 and later for Windows virtual machines, vCenter Server 7.0 Update 2 and later for Linux virtual machines.
  - Virtual machine encryption (to encrypt the virtual machine home files).
  - Key provider configured for vCenter Server. See Comparison of vSphere Key Providers.
- Guest OS support:
  - Linux
  - Windows Server 2008 and later
  - Windows 7 and later

Differences between a Hardware TPM and a Virtual TPM

You use a hardware Trusted Platform Module (TPM) to provide secure storage of credentials or keys. A vTPM performs the same functions as a TPM, but it performs cryptographic coprocessor capabilities in software. A vTPM uses the `.nvram` file, which is encrypted using virtual machine encryption, as its secure storage.

A hardware TPM includes a preloaded key called the Endorsement Key (EK). The EK has a private and public key. The EK provides the TPM with a unique identity. For a vTPM, this key is provided either by the VMware Certificate Authority (VMCA) or by a third-party Certificate Authority (CA). After the vTPM uses a key, it is typically not changed because doing so invalidates sensitive information stored in the vTPM. The vTPM does not contact the third-party CA at any time.

Create a Virtual Machine with a Virtual Trusted Platform Module

You can add a Virtual Trusted Platform Module (vTPM) when you create a virtual machine to provide enhanced security to the guest operating system. You must create a key provider before you can add a vTPM.

The VMware virtual TPM is compatible with TPM 2.0 and creates a TPM-enabled virtual chip for use by the virtual machine and the guest OS it hosts.

Prerequisites

- Ensure that your vSphere environment is configured with a key provider. See the following for more information:
  - Configuring vSphere Trust Authority
  - Chapter 7 Configuring and Managing a Standard Key Provider
  - Chapter 8 Configuring and Managing vSphere Native Key Provider
- The guest OS you use can be Windows Server 2008 and later, Windows 7 and later, or Linux.
- The ESXi hosts running in your environment must be ESXi 6.7 or later (Windows guest OS), or 7.0 Update 2 (Linux guest OS).
- The virtual machine must use EFI firmware.
- Verify that you have the required privileges:
  - **Cryptographic operations.Clone**
  - **Cryptographic operations.Encrypt**
  - **Cryptographic operations.Encrypt new**
  - **Cryptographic operations.Migrate**
  - **Cryptographic operations.Register VM**

Note After creating a virtual machine with a vTPM, the **Cryptographic operations.Direct Access** privilege is required to open a console session.

Procedure

1 Connect to vCenter Server by using the vSphere Client.

2 Select an object in the inventory that is a valid parent object of a virtual machine, for example, an ESXi host or a cluster.

3 Right-click the object, select **New Virtual Machine**, and follow the prompts to create a virtual machine.

Option	Action
Select a creation type	Create a new virtual machine.
Select a name and folder	Specify a name and target location.
Select a compute resource	Specify an object for which you have privileges to create a virtual machine. See Prerequisites and Required Privileges for Encryption Tasks.
Select storage	Select a compatible datastore.
Select compatibility	You must select **ESXi 6.7 and later** for Windows guest OS, or **ESXi 7.0 U2** and later for Linux guest OS.
Select a guest OS	Select Windows or Linux for use as the guest OS.
Customize hardware	Click **Add New Device** and select **Trusted Platform Module**. You can further customize the hardware, for example, by changing disk size or CPU.
Ready to complete	Review the information and click **Finish**.

Results

The vTPM-enabled virtual machine appears in your inventory as specified.

Enable Virtual Trusted Platform Module for an Existing Virtual Machine

You can add a Virtual Trusted Platform Module (vTPM) to an existing virtual machine to provide enhanced security to the guest operating system. You must create a key provider before you can add a vTPM.

The VMware virtual TPM is compatible with TPM 2.0, and creates a TPM-enabled virtual chip for use by the virtual machine and the guest OS it hosts.

Prerequisites

- Ensure that your vSphere environment is configured for a key provider. See the following for more information:
  - Configuring vSphere Trust Authority
  - Chapter 7 Configuring and Managing a Standard Key Provider
  - Chapter 8 Configuring and Managing vSphere Native Key Provider
- The guest OS you use can be Windows Server 2008 and later, Windows 7 and later, or Linux.
- Verify that the virtual machine is turned off.

- The ESXi hosts running in your environment must be ESXi 6.7 or later (Windows guest OS), or 7.0 Update 2 (Linux guest OS).
- The virtual machine must use EFI firmware.
- Verify that you have the required privileges:
  - **Cryptographic operations.Clone**
  - **Cryptographic operations.Encrypt**
  - **Cryptographic operations.Encrypt new**
  - **Cryptographic operations.Migrate**
  - **Cryptographic operations.Register VM**
  - **Virtual machine.Configuration.Add or remove device**

Note After adding a vTPM to a virtual machine, the **Cryptographic operations.Direct Access** privilege is required to open a console session.

Procedure

1. Connect to vCenter Server by using the vSphere Client.
2. Right-click the virtual machine in the inventory that you want to modify and select **Edit Settings**.
3. In the **Edit Settings** dialog box, click **Add New Device** and select **Trusted Platform Module**.
4. Click **OK**.

   The virtual machine **Summary** tab now includes Virtual Trusted Platform Module in the **VM Hardware** pane.

Remove Virtual Trusted Platform Module from a Virtual Machine

You can remove Virtual Trusted Platform Module (vTPM) security from a virtual machine.

Removing a vTPM device causes all encrypted information on the virtual machine to become unrecoverable. Before removing a vTPM from a virtual machine, disable any applications in the Guest OS that use the vTPM device, such as BitLocker. Failure to do so can cause the virtual machine not to boot. Also, you cannot remove a vTPM from a virtual machine that contains snapshots.

Prerequisites

- Ensure that the virtual machine is powered off.
- Verify that you have the required privileges: **Virtual machine.Configuration.Add or remove device** and **Cryptographic operations.Decrypt**

Procedure

1 Connect to vCenter Server by using the vSphere Client.

2 Right-click the virtual machine in the inventory that you want to modify and select **Edit Settings**.

3 In the **Edit Settings** dialog box, locate the Trusted Platform Module entry in the **Virtual Hardware** tab.

4 Move your pointer over the device and click the **Remove** icon.

This icon appears only for the virtual hardware that you can safely remove.

5 Click **Delete** to confirm you want to remove the device.

The vTPM device is marked for removal.

6 Click **OK**.

Verify that the Virtual Trusted Platform Module entry no longer appears in the virtual machine **Summary** tab in the **VM Hardware** pane.

Identify Virtual Trusted Platform Module Enabled Virtual Machines

You can identify which of your virtual machines are enabled to use a Virtual Trusted Platform Module (vTPM).

You can generate a list of all virtual machines in your inventory showing virtual machine name, operating system, and vTPM status. You can also export this list to a CSV file for use in compliance audits.

Procedure

1 Connect to vCenter Server by using the vSphere Client.

2 Select a vCenter Server instance, a host, or a cluster.

3 Click the **VMs** tab and click **Virtual Machines**.

4 To view all virtual machines on which a TPM is enabled, click the three-bar **Column Selector** in the lower left corner and select **TPM**.

The TPM column displays "Present" for virtual machines on which a TPM is enabled. Virtual machines without a TPM are listed as "Not present."

5 You can export the contents of an inventory list view to a CSV file.

   a Click **Export** at the bottom-right corner of a list view.

     The Export List Contents dialog box opens and lists the available options for inclusion in the CSV file.

   b Select whether you want all rows or your current selection of rows to be listed in the CSV file.

   c From the available options, select the columns you want listed in the CSV file.

   d Click **Export**.

     The CSV file is generated and available for download.

View Virtual Trusted Platform Module Device Certificates

Virtual Trusted Platform Module (vTPM) devices are pre-configured with default certificates, which you can review.

Prerequisites

You must have a vTPM-enabled virtual machine in your environment.

Procedure

1 Connect to vCenter Server by using the vSphere Client.

2 Select an object in the inventory that is a valid parent object of a virtual machine, for example, an ESXi host or a cluster.

3 Click **VMs** and click **Virtual Machines**.

4 Select the vTPM-enabled virtual machine whose certificate information you want to view.

   If necessary, click the three-bar **Column Selector** in the lower left corner and select **TPM** to display virtual machines with a TPM "Present."

5 Click the **Configure** tab.

6 Under **TPM**, select **Certificates**.

7 Select the certificate and view its information.

8 (Optional) To export the certificate information, click **Export**.

   The certificate is saved to disk.

What to do next

You can replace the default certificate with a certificate issued by a third-party certificate authority (CA). See Export and Replace Virtual Trusted Platform Module Device Certificates.

Export and Replace Virtual Trusted Platform Module Device Certificates

You can replace the default certificate that comes with a Virtual Trusted Platform Module (vTPM) device.

Prerequisites

You must have a vTPM-enabled virtual machine in your environment.

Procedure

1. Connect to vCenter Server by using the vSphere Client.
2. Select an object in the inventory that is a valid parent object of a virtual machine, for example, an ESXi host or a cluster.
3. Select the vTPM-enabled virtual machine in the inventory whose certificate information you want to replace.
4. Click the **Configure** tab.
5. Under **TPM** select **Signing Requests**.
6. Select a certificate.
7. To export the certificate information, click **Export**.

   The certificate is saved to disk.
8. Get a certificate issued by a third-party certificate authority (CA) against the certificate signing request (CSR) you exported.

   You can use any CA that you might have in your IT environment.
9. When you have the new certificate, replace the existing certificate.
   a. Right-click the virtual machine in the inventory whose certificate you want to replace and select **Edit Settings**.
   b. In the **Edit Settings** dialog box, expand **Security Devices**, then expand **Trusted Platform Module**.

      The certificates appear.
   c. Click **Replace** for the certificate you want to replace.

      The **File Upload** dialog box appears.
   d. On your local machine, locate the new certificate and upload it.

      The new certificate replaces the default certificate that came with your vTPM device.
   e. The certificate name is updated in the virtual machine **Summary** tab under the **Virtual Trusted Platform Module** list.

Securing Windows Guest Operating Systems with Virtualization-based Security

12

Starting with vSphere 6.7, you can enable Microsoft virtualization-based security (VBS) on supported Windows guest operating systems.

Microsoft VBS, a feature of Windows 10 and Windows Server 2016 operating systems, uses hardware and software virtualization to enhance system security by creating an isolated, hypervisor-restricted, specialized subsystem.

VBS permits you to use the following Windows security features to harden your system and isolate key system and user secrets from being compromised:

- Credential Guard: Aims to isolate and harden key system and user secrets against compromise.
- Device Guard: Provides a set of features designed to work together to prevent and eliminate malware from running on a Windows system.
- Configurable Code Integrity: Ensures that only trusted code runs from the boot loader onwards.

See the topic on virtualization-based security in the Microsoft documentation for more information.

After you enable VBS for a virtual machine through vCenter Server, you enable VBS within the Windows guest operating system.

Read the following topics next:

Virtualization-based Security Best Practices

Follow best practices for virtualization-based security (VBS) to maximize security and manageability of your Windows guest operating system environment.

Avoid problems by following these best practices.

VBS Hardware

Use the following hardware for VBS:

- Intel
  - Haswell CPU or later. For best performance, use the Skylake-EP CPU or later.
  - The Ivy Bridge CPU is acceptable.
  - The Sandy Bridge CPU might cause some slow performance.
- AMD
  - Zen 2 series CPUs (Rome) or later.
  - Older CPUs might cause some slow performance.

The mitigations for the Machine Check Exception on Page Size Change Intel CPU vulnerability can impact guest OS performance negatively when VBS is in use. For more information, see the VMware KB article at https://kb.vmware.com/kb/76050.

Windows Guest OS Compatibility

On Intel, VBS is supported for Windows 10 and Windows Server 2016 and later virtual machines, although Windows Server 2016 versions 1607 and 1703 require patches. Check the Microsoft documentation for ESXi host hardware compatibility. Using Intel CPUs for VBS requires vSphere 6.7 or later and hardware version 14.

On AMD, VBS is supported on Windows 10, version 1809, and Windows 2019 and later virtual machines. Using AMD CPUs for VBS requires vSphere 7.0 Update 2 or later and hardware version 19.

Initially, Windows 10 required that you enable Hyper-V for VBS. Enabling Hyper-V is not required for Windows 10. The same applies to Windows Server 2016 and later. Consult the current Microsoft documentation and the *VMware vSphere Release Notes* for more information.

Unsupported VMware Features on VBS

The following features are not supported in a virtual machine when VBS is enabled:

- Fault tolerance
- PCI passthrough
- Hot add of CPU or memory

Installation and Upgrade Caveats with VBS

Before you configure VBS, understand the following installation and upgrade caveats:

- New virtual machines configured for Windows 10 and Windows Server 2016 and later on virtual hardware versions less than version 14 are created using Legacy BIOS by default. You must reinstall the guest operating system after changing the virtual machine's firmware type from Legacy BIOS to UEFI.
- If you plan to migrate your virtual machines from previous vSphere releases to vSphere 6.7 or greater, and enable VBS on your virtual machines, use UEFI to avoid having to reinstall the operating system.

Enable Virtualization-based Security on a Virtual Machine

You can enable Microsoft virtualization-based security (VBS) for supported Windows guest operating systems at the same time you create a virtual machine.

Enabling VBS is a process that involves first enabling VBS in the virtual machine then enabling VBS in the Windows guest OS.

Prerequisites

See Virtualization-based Security Best Practices for acceptable CPUs.

Using Intel CPUs for VBS requires vSphere 6.7 or later. Create a virtual machine that uses hardware version 14 or later and one of the following supported guest operating systems:

- Windows 10 (64 bit) or later releases
- Windows Server 2016 (64 bit) or later releases

Using AMD CPUs for VBS requires vSphere 7.0 Update 2 or later. Create a virtual machine that uses hardware version 19 or later and one of the following supported guest operating systems:

- Windows 10 (64 bit), version 1809 or later releases
- Windows Server 2019 (64 bit) or later releases

Ensure that you install the latest patches for Windows 10, version 1809, and Windows Server 2019, before enabling VBS.

For more information about activating VBS on virtual machines on AMD platforms, see the VMware KB article at https://kb.vmware.com/s/article/89880.

Procedure

1. Connect to vCenter Server by using the vSphere Client.
2. Select an object in the inventory that is a valid parent object of a virtual machine, for example, an ESXi host or a cluster.

3 Right-click the object, select **New Virtual Machine**, and follow the prompts to create a virtual machine.

Option	Action
Select a creation type	Create a virtual machine.
Select a name and folder	Specify a name and target location.
Select a compute resource	Specify an object for which you have privileges to create virtual machines.
Select storage	In the VM storage policy, select the storage policy. Select a compatible datastore.
Select compatibility	Intel CPU: Ensure that **ESXi 6.7 and later** is selected. AMD CPU: Ensure that **ESXi 7.0 U2 and later** is selected.
Select a guest OS	Select the Windows guest operating system option that best corresponds to operating system release. Select the **Enable Windows Virtualization Based Security** check box.
Customize hardware	Customize the hardware, for example, by changing disk size or CPU.
Ready to complete	Review the information and click **Finish**.

Results

Once the virtual machine is created, confirm that its **Summary** tab displays "VBS true" in the Guest OS description.

What to do next

See Enable Virtualization-based Security on the Guest Operating System.

Enable Virtualization-based Security on an Existing Virtual Machine

You can enable Microsoft virtualization-based security (VBS) on existing virtual machines for supported Windows guest operating systems.

Enabling VBS is a process that involves first enabling VBS in the virtual machine then enabling VBS in the guest OS.

Note New virtual machines configured for Windows 10, Windows Server 2016, and Windows Server 2019 on hardware versions less than version 14 are created using Legacy BIOS by default. If you change the virtual machine's firmware type from Legacy BIOS to UEFI, you must reinstall the guest operating system.

Prerequisites

See Virtualization-based Security Best Practices for acceptable CPUs.

Using Intel CPUs for VBS requires vSphere 6.7 or later. The virtual machine must have been created using hardware version 14 or later and one of the following supported guest operating systems:

- Windows 10 (64 bit) or later releases
- Windows Server 2016 (64 bit) or later releases

Using AMD CPUs for VBS requires vSphere 7.0 Update 2 or later. The virtual machine must have been created using hardware version 19 or later and one of the following supported guest operating systems:

- Windows 10 (64 bit), version 1809 or later releases
- Windows Server 2019 (64 bit) or later releases

Ensure that you install the latest patches for Windows 10, version 1809, and Windows Server 2019, before enabling VBS.

For more information about activating VBS on virtual machines on AMD platforms, see the VMware KB article at https://kb.vmware.com/s/article/89880.

Procedure

1. In the vSphere Client, browse to the virtual machine.
2. Right-click the virtual machine and select **Edit Settings**.
3. Click the **VM Options** tab.
4. Select the **Enable** check box for Virtualization Based Security.
5. Click **OK**.

Results

Confirm that the virtual machine's **Summary** tab displays "VBS true" in the Guest OS description.

What to do next

See Enable Virtualization-based Security on the Guest Operating System.

Enable Virtualization-based Security on the Guest Operating System

You can enable Microsoft virtualization-based security (VBS) for supported Windows guest operating systems.

You enable VBS from within the Windows Guest OS. Windows configures and enforces VBS through a Group Policy Object (GPO). The GPO gives you the ability to turn off and on the various services, such as Secure Boot, Device Guard, and Credential Guard, that VBS offers. Certain Windows versions also require you to perform the additional step of enabling the Hyper-V platform.

See Microsoft's documentation about deploying Device Guard to enable virtualization-based security for details.

Prerequisites

- Ensure that virtualization-based security has been enabled on the virtual machine.

Procedure

1. In Microsoft Windows, edit the group policy to turn on VBS and choose other VBS-related security options.
2. (Optional) For Microsoft Windows versions less than Redstone 4, in the Windows Features control panel, enable the Hyper-V platform.
3. Reboot the guest operating system.

Disable Virtualization-based Security

If you no longer use virtualization-based security (VBS) with a virtual machine, you can disable VBS. When you disable VBS for the virtual machine, the Windows VBS options remain unchanged but might induce performance issues. Before disabling VBS on the virtual machine, disable VBS options within Windows.

Prerequisites

Ensure that the virtual machine is powered off.

Procedure

1. In the vSphere Client, browse to the VBS-enabled virtual machine.

   See Identify VBS-Enabled Virtual Machines for help in locating VBS-enabled virtual machines.
2. Right-click the virtual machine and select **Edit Settings**.
3. Click **VM Options**.
4. Deselect the **Enable** check box for Virtualization Based Security.

   A message reminds you to disable VBS in the guest OS.
5. Click **OK**.
6. Verify that the virtual machine's **Summary** tab no longer displays "VBS true" in the Guest OS description.

Identify VBS-Enabled Virtual Machines

You can identify which of your virtual machines have VBS enabled, for reporting and compliance purposes.

Procedure

1 Connect to vCenter Server by using the vSphere Client.

2 Select a vCenter Server instance, a data center, or a host in the inventory.

3 Click the **VMs** tab and click **Virtual Machines**.

4 To show the **VBS** column, click the three-bar **Column Selector** in the lower left corner and select the **VBS** check box.

5 Scan for "Present" in the **VBS** column.

Securing vSphere Networking

13

Securing vSphere Networking is an essential part of protecting your environment. You secure different vSphere components in different ways. See the *vSphere Networking* documentation for detailed information about networking in the vSphere environment.

Read the following topics next:

Introduction to vSphere Network Security

Network security in the vSphere environment shares many characteristics of securing a physical network environment, but also includes some characteristics that apply only to virtual machines.

Firewalls

Add firewall protection to your virtual network by installing and configuring host-based firewalls on some or all its VMs.

For efficiency, you can set up private virtual machine Ethernet networks or virtual networks. With virtual networks, you install a host-based firewall on a VM at the head of the virtual network. This firewall serves as a protective buffer between the physical network adapter and the remaining VMs in the virtual network.

Host-based firewalls can slow performance. Balance your security needs against performance goals before you install host-based firewalls on VMs elsewhere in the virtual network.

See Securing the Network with Firewalls.

Segmentation

Keep different virtual machine zones within a host on different network segments. If you isolate each virtual machine zone on its own network segment, you minimize the risk of data leakage from one zone to the next. Segmentation prevents various threats, including Address Resolution Protocol (ARP) spoofing. With ARP spoofing, an attacker manipulates the ARP table to remap MAC and IP addresses, and gains access to network traffic to and from a host. Attackers use ARP spoofing to generate man in the middle (MITM) attacks, perform denial of service (DoS) attacks, hijack the target system, and otherwise disrupt the virtual network.

Planning segmentation carefully lowers the chances of packet transmissions between virtual machine zones. Segmentation therefore prevents sniffing attacks that require sending network traffic to the victim. Also, an attacker cannot use a nonsecure service in one virtual machine zone to access other virtual machine zones in the host. You can implement segmentation by using one of two approaches.

- Use separate physical network adapters for virtual machine zones to ensure that the zones are isolated. Maintaining separate physical network adapters for virtual machine zones is probably the most secure method. After the initial segment creation. This approach is less prone to misconfiguration.
- Set up virtual local area networks (VLANs) to help safeguard your network. VLANs provide almost all the security benefits inherent in implementing physically separate networks without the hardware overhead. VLANs can save you the cost of deploying and maintaining additional devices, cabling, and so on. See Securing Virtual Machines with VLANs.

Preventing Unauthorized Access

Requirements for securing VMs are often the same as requirements for securing physical machines.

- If a virtual machine network is connected to a physical network, it can be subject to breaches just like a network that consists of physical machines.
- Even if you do not connect a VM to the physical network, the VM can be attacked by other VMs.

VMs are isolated from each other. One VM cannot read or write another VM's memory, access its data, use its applications, and so forth. However, within the network, any VM or group of VMs can still be the target of unauthorized access from other VMs. Protect your VMs from such unauthorized access.

For additional information about protecting VMs, see the NIST document titled " Secure Virtual Network Configuration for Virtual Machine (VM) Protection" at:

https://csrc.nist.gov/publications/detail/sp/800-125b/final

Securing the Network with Firewalls

Security administrators use firewalls to safeguard the network or selected components in the network from intrusion.

Firewalls control access to devices within their perimeter by closing all ports except for ports that the administrator explicitly or implicitly designates as authorized. The ports that administrators open allow traffic between devices on different sides of the firewall.

Important The ESXi firewall in ESXi 5.5 and later does not allow per-network filtering of vMotion traffic. Therefore, you must install rules on your external firewall to ensure that no incoming connections can be made to the vMotion socket.

In a virtual machine environment, you can plan the layout for firewalls between components.

- Firewalls between physical machines such as vCenter Server systems and ESXi hosts.
- Firewalls between one virtual machine and another, for example, between a virtual machine acting as an external Web server and a virtual machine connected to your company's internal network.
- Firewalls between a physical machine and a virtual machine, such as when you place a firewall between a physical network adapter card and a virtual machine.

How you use firewalls in your ESXi configuration is based on how you plan to use the network and how secure any given component has to be. For example, if you create a virtual network where each virtual machine is dedicated to running a different benchmark test suite for the same department, the risk of unwanted access from one virtual machine to the next is minimal. Therefore, a configuration where firewalls are present between the virtual machines is not necessary. However, to prevent interruption of a test run from an outside host, you can configure a firewall at the entry point of the virtual network to protect the entire set of virtual machines.

For the list of all supported ports and protocols in VMware products, including vSphere and vSAN, see the VMware Ports and Protocols Tool™ at https://ports.vmware.com/. You can search ports by VMware product, create a customized list of ports, and print or save port lists.

Firewalls for Configurations with vCenter Server

If you access ESXi hosts through vCenter Server, you typically protect vCenter Server using a firewall.

Firewalls must be present at entry points. A firewall might lie between the clients and vCenter Server or vCenter Server and the clients can both be behind a firewall.

For the list of all supported ports and protocols in VMware products, including vSphere and vSAN, see the VMware Ports and Protocols Tool™ at https://ports.vmware.com/. You can search ports by VMware product, create a customized list of ports, and print or save port lists.

Networks configured with vCenter Server can receive communications through the vSphere Client, other UI clients, or clients that use the vSphere API. During normal operation, vCenter Server listens for data from its managed hosts and clients on designated ports. vCenter Server also assumes that its managed hosts listen for data from vCenter Server on designated ports. If a firewall is present between any of these elements, you must ensure that the firewall has open ports to support data transfer.

You might also include firewalls at other access points in the network, depending on the network usage and on the level of security that clients require. Select the locations for your firewalls based on the security risks for your network configuration. The following firewall locations are commonly used.

- Between the vSphere Client or a third-party network-management client and vCenter Server.
- If your users access virtual machines through a Web browser, between the Web browser and the ESXi host.
- If your users access virtual machines through the vSphere Client, between the vSphere Client and the ESXi host. This connection is in addition to the connection between the vSphere Client and vCenter Server, and it requires a different port.
- Between vCenter Server and the ESXi hosts.
- Between the ESXi hosts in your network. Although traffic between hosts is usually considered trusted, you can add firewalls between them if you are concerned about security breaches from machine to machine.

  If you add firewalls between ESXi hosts and plan to migrate virtual machines between them, open ports in any firewall that divides the source host from the target hosts.
- Between the ESXi hosts and network storage such as NFS or iSCSI storage. These ports are not specific to VMware. Configure them according to the specifications for your network.

Connecting to vCenter Server Through a Firewall

Open TCP port 443 in the firewall to enable vCenter Server to receive data.

By default vCenter Server uses TCP port 443 to listen for data from its clients. If you have a firewall between vCenter Server and its clients, you must configure a connection through which vCenter Server can receive data from the clients. Firewall configuration depends on what is used at your site, ask your local firewall system administrator for information.

Connecting ESXi Hosts Through Firewalls

If you have a firewall between your ESXi hosts and vCenter Server, ensure that the managed hosts can receive data.

To configure a connection for receiving data, open ports for traffic from services such as vSphere High Availability, vMotion, and vSphere Fault Tolerance. See ESXi Firewall Configuration for a discussion of configuration files, vSphere Client access, and firewall commands. For a list of ports, see the VMware Ports and Protocols Tool™ at https://ports.vmware.com.

Firewalls for Configurations Without vCenter Server

If your environment does not include vCenter Server, clients can connect directly to the ESXi network.

You can connect to a standalone ESXi host in several ways.

- VMware Host Client
- ESXCLI interface
- vSphere Web Services SDK or vSphere Automation SDKs
- Third-party clients

The firewall requirements for standalone hosts are similar to requirements when a vCenter Server is present.

- Use a firewall to protect your ESXi layer or, depending on your configuration, your clients, and the ESXi layer. This firewall provides basic protection for your network.
- Licensing in this type of configuration is part of the ESXi package that you install on each of the hosts. Because licensing is resident to ESXi, a separate License Server with a firewall is not required.

You can configure firewall ports using ESXCLI or using the VMware Host Client. See *vSphere Single Host Management - VMware Host Client*.

Connecting to the Virtual Machine Console Through a Firewall

Certain ports must be open for user and administrator communication with the virtual machine console. Which ports must be open depends on the type of virtual machine console, and on whether you connect through vCenter Server with the vSphere Client or directly to the ESXi host from the VMware Host Client.

For more information about ports, purpose, and classification (incoming, outgoing, or bidirectional), see the VMware Ports and Protocols Tool™ at https://ports.vmware.com.

Connecting to a Browser-Based Virtual Machine Console Through the vSphere Client

When you are connecting with the vSphere Client, you always connect to the vCenter Server system that manages the ESXi host, and access the virtual machine console from there.

If you are using the vSphere Client and connecting to a browser-based virtual machine console, the following access must be possible:

- The firewall must allow vSphere Client to access vCenter Server on port 443.
- The firewall must allow vCenter Server to access the ESXi host on port 902.

Connecting to a VMware Remote Console Through the vSphere Client

If you are using the vSphere Client and connecting to a VMware Remote Console (VMRC), the following access must be possible:

- The firewall must allow the vSphere Client to access vCenter Server on port 443.
- The firewall must allow the VMRC to access vCenter Server on port 443 and to access the ESXi host on port 902 for VMRC versions before 11.0, and port 443 for VMRC version 11.0 and greater. For more information about VMRC version 11.0 and ESXi port requirements, see the VMware knowledge base article at https://kb.vmware.com/s/article/76672.

Connecting to ESXi Hosts Directly with the VMware Host Client

You can use the VMware Host Client virtual machine console if you connect directly to an ESXi host.

Note Do not use the VMware Host Client to connect directly to hosts that are managed by a vCenter Server system. If you make changes to such hosts from the VMware Host Client, instability in your environment results.

The firewall must allow access to the ESXi host on ports 443 and 902.

The VMware Host Client uses port 902 to provide a connection for guest operating system MKS activities on virtual machines. It is through this port that users interact with the guest operating systems and applications of the virtual machine. VMware does not support configuring a different port for this function.

Secure the Physical Switch

Secure the physical switch on each ESXi host to prevent attackers from gaining access to the host and its virtual machines.

For best protection of your hosts, ensure that physical switch ports are configured with spanning tree disabled and ensure that the non-negotiate option is configured for trunk links between external physical switches and virtual switches in Virtual Switch Tagging (VST) mode.

Procedure

1 Log in to the physical switch and ensure that spanning tree protocol is disabled or that Port Fast is configured for all physical switch ports that are connected to ESXi hosts.

2 For virtual machines that perform bridging or routing, check periodically that the first upstream physical switch port is configured with BPDU Guard and Port Fast disabled and with spanning tree protocol enabled.

   In vSphere 5.1 and later, to prevent the physical switch from potential Denial of Service (DoS) attacks, you can turn on the guest BPDU filter on the ESXi hosts.

3 Log in to the physical switch and ensure that Dynamic Trunking Protocol (DTP) is not enabled on the physical switch ports that are connected to the ESXi hosts.

4 Routinely check physical switch ports to ensure that they are properly configured as trunk ports if connected to virtual switch VLAN trunking ports.

Securing Standard Switch Ports with Security Policies

The VMkernel port group or virtual machine port group on a standard switch has a configurable security policy. The security policy determines how strongly you enforce protection against impersonation and interception attacks on VMs.

Just like physical network adapters, virtual machine network adapters can impersonate another VM. Impersonation is a security risk.

- A VM can send frames that appear to be from a different machine so that it can receive network frames that are intended for that machine.
- A virtual machine network adapter can be configured so that it receives frames targeted for other machines

When you add a VMkernel port group or virtual machine port group to a standard switch, ESXi configures a security policy for the ports in the group. You can use this security policy to ensure that the host prevents the guest operating systems of its VMs from impersonating other machines on the network. The guest operating system that might attempt impersonation does not detect that the impersonation was prevented.

The security policy determines how strongly you enforce protection against impersonation and interception attacks on VMs. To correctly use the settings in the security profile, see the Security Policy section in the *vSphere Networking* publication. This section explains:

- How VM network adapters control transmissions.
- How attacks are staged at this level

Securing vSphere Standard Switches

You can secure standard switch traffic against Layer 2 attacks by restricting some of the MAC address modes of the VM network adapters.

Each VM network adapter has an initial MAC address and an effective MAC address.

Initial MAC address

The initial MAC address is assigned when the adapter is created. Although the initial MAC address can be reconfigured from outside the guest operating system, it cannot be changed by the guest operating system.

Effective MAC address

Each adapter has an effective MAC address that filters out incoming network traffic with a destination MAC address that is different from the effective MAC address. The guest operating system is responsible for setting the effective MAC address and typically matches the effective MAC address to the initial MAC address.

Upon creating a VM network adapter, the effective MAC address and initial MAC address are the same. The guest operating system can alter the effective MAC address to another value at any time. If an operating system changes the effective MAC address, its network adapter receives network traffic that is destined for the new MAC address.

When sending packets through a network adapter, the guest operating system typically places its own adapter effective MAC address in the source MAC address field of the Ethernet frames. It places the MAC address for the receiving network adapter in the destination MAC address field. The receiving adapter accepts packets only if the destination MAC address in the packet matches its own effective MAC address.

An operating system can send frames with an impersonated source MAC address. An operating system can therefore impersonate a network adapter that the receiving network authorizes, and stage malicious attacks on the devices in a network.

Protect virtual traffic against impersonation and interception Layer 2 attacks by configuring a security policy on port groups or ports.

The security policy on distributed port groups and ports includes the following options:

- MAC address changes (see MAC Address Changes)
- Promiscuous mode (see Promiscuous Mode Operation)
- Forged transmits (see Forged Transmits)

You can view and change the default settings by selecting the virtual switch associated with the host from the vSphere Client. See *vSphere Networking* documentation.

MAC Address Changes

The security policy of a virtual switch includes a **MAC address changes** option. This option allows virtual machines to receive frames with a Mac Address that is different from the one configured in the VMX.

When the **Mac address changes** option is set to **Accept**, ESXi accepts requests to change the effective MAC address of a virtual machine to a different address than the initial MAC address.

When the **Mac address changes** option is set to **Reject**, ESXi does not honor requests to change the effective MAC address of a virtual machine to a different address than the initial MAC address. This setting protects the host against MAC impersonation. The port that the virtual machine adapter used to send the request is disabled and the virtual machine adapter does not receive any more frames until the effective MAC address matches the initial MAC address. The guest operating system does not detect that the MAC address change request was not honored.

Note The iSCSI initiator relies on being able to get MAC address changes from certain types of storage. If you are using ESXi iSCSI with iSCSI storage, set the **MAC address changes** option to **Accept**.

In some situations, you can have a legitimate need for more than one adapter to have the same MAC address on a network, for example, if you are using Microsoft Network Load Balancing in unicast mode. When Microsoft Network Load Balancing is used in the standard multicast mode, adapters do not share MAC addresses.

Note Starting in vSphere 7.0, the defaults for **Forged transmits** and **MAC address changes** have been changed to Reject instead of Accept. Contact your storage vendor to validate.

Forged Transmits

The **Forged transmits** option affects traffic that is transmitted from a virtual machine.

When the **Forged transmits** option is set to **Accept**, ESXi does not compare source and effective MAC addresses.

To protect against MAC impersonation, you can set the **Forged transmits** option to **Reject**. If you do, the host compares the source MAC address being transmitted by the guest operating system with the effective MAC address for its virtual machine adapter to see if they match. If the addresses do not match, the ESXi host drops the packet.

The guest operating system does not detect that its virtual machine adapter cannot send packets by using the impersonated MAC address. The ESXi host intercepts any packets with impersonated addresses before they are delivered, and the guest operating system might assume that the packets are dropped.

Note Starting in vSphere 7.0, the defaults for **Forged transmits** and **MAC address changes** have been changed to Reject instead of Accept.

Promiscuous Mode Operation

Promiscuous mode eliminates any reception filtering that the virtual machine adapter performs so that the guest operating system receives all traffic observed on the wire. By default, the virtual machine adapter cannot operate in promiscuous mode.

Although promiscuous mode can be useful for tracking network activity, it is an insecure mode of operation, because any adapter in promiscuous mode has access to the packets even if some of the packets are received only by a particular network adapter. This means that an administrator or root user within a virtual machine can potentially view traffic destined for other guest or host operating systems.

See the topic on configuring the security policy for a vSphere Standard Switch or Standard Port Group in the *vSphere Networking* documentation for information about configuring the virtual machine adapter for promiscuous mode.

Note In some situations, you might have a legitimate reason to configure a standard or a distributed virtual switch to operate in promiscuous mode, for example, if you are running network intrusion detection software or a packet sniffer.

Standard Switch Protection and VLANs

VMware standard switches provide safeguards against certain threats to VLAN security. Because of the way that standard switches are designed, they protect VLANs against a variety of attacks, many of which involve VLAN hopping.

Having this protection does not guarantee that your virtual machine configuration is invulnerable to other types of attacks. For example, standard switches do not protect the physical network against these attacks; they protect only the virtual network.

Standard switches and VLANs can protect against the following types of attacks.

MAC flooding

Floods a switch with packets that contain MAC addresses tagged as having come from different sources. Many switches use a content-addressable memory table to learn and store the source address for each packet. When the table is full, the switch can enter a fully open state in which every incoming packet is broadcast on all ports, letting the attacker see all of the switch's traffic. This state might result in packet leakage across VLANs.

Although VMware standard switches store a MAC address table, they do not get the MAC addresses from observable traffic and are not vulnerable to this type of attack.

802.1q and ISL tagging attacks

Force a switch to redirect frames from one VLAN to another by tricking the switch into acting as a trunk and broadcasting the traffic to other VLANs.

VMware standard switches do not perform the dynamic trunking required for this type of attack and, therefore, are not vulnerable.

Double-encapsulation attacks

Occur when an attacker creates a double-encapsulated packet in which the VLAN identifier in the inner tag is different from the VLAN identifier in the outer tag. For backward compatibility, native VLANs strip the outer tag from transmitted packets unless configured

to do otherwise. When a native VLAN switch strips the outer tag, only the inner tag is left, and that inner tag routes the packet to a different VLAN than the one identified in the now-missing outer tag.

VMware standard switches drop any double-encapsulated frames that a virtual machine attempts to send on a port configured for a specific VLAN. Therefore, they are not vulnerable to this type of attack.

Multicast brute-force attacks

Involve sending large numbers of multicast frames to a known VLAN almost simultaneously to overload the switch so that it mistakenly allows some of the frames to broadcast to other VLANs.

VMware standard switches do not allow frames to leave their correct broadcast domain (VLAN) and are not vulnerable to this type of attack.

Spanning-tree attacks

Target Spanning-Tree Protocol (STP), which is used to control bridging between parts of the LAN. The attacker sends Bridge Protocol Data Unit (BPDU) packets that attempt to change the network topology, establishing themselves as the root bridge. As the root bridge, the attacker can sniff the contents of transmitted frames.

VMware standard switches do not support STP and are not vulnerable to this type of attack.

Random frame attacks

Involve sending large numbers of packets in which the source and destination addresses stay the same, but in which fields are randomly changed in length, type, or content. The goal of this attack is to force packets to be mistakenly rerouted to a different VLAN.

VMware standard switches are not vulnerable to this type of attack.

Because new security threats develop over time, do not consider this an exhaustive list of attacks. Regularly check VMware security resources on the Web to learn about security, recent security alerts, and VMware security tactics.

Secure vSphere Distributed Switches and Distributed Port Groups

Administrators have several options for securing vSphere Distributed Switches in their vSphere environment.

The same rules apply for VLANs in a vSphere Distributed Switch as they do in a standard switch. For more information, see Standard Switch Protection and VLANs.

Procedure

1 For distributed port groups with static binding, disable the Auto Expand feature.

Auto Expand is enabled by default in vSphere 5.1 and later.

To disable Auto Expand, configure the `autoExpand` property under the distributed port group with the vSphere Web Services SDK or with a command-line interface. See the *vSphere Web Services SDK* documentation.

2 Ensure that all private VLAN IDs of any vSphere Distributed Switch are fully documented.

3 If you are using VLAN tagging on a dvPortgroup, VLAN IDs must correspond to the IDs on external VLAN-aware upstream switches. If VLAN IDs are not tracked correctly, mistaken reuse of IDs might allow unintended traffic. Similarly, wrong or missing VLAN IDs might lead to traffic not passing between physical and virtual machines.

4 Ensure that no unused ports exist on a virtual port group associated with a vSphere Distributed Switch.

5 Label all vSphere Distributed Switches.

vSphere Distributed Switches associated with an ESXi host require a text box for the name of the switch. This label serves as a functional descriptor for the switch, just like the host name associated with a physical switch. The label on the vSphere Distributed Switch indicates the function or the IP subnet of the switch. For example, you can label the switch as internal to indicate that it is only for internal networking on a virtual machine's private virtual switch. No traffic goes over physical network adapters.

6 Disable network health check for your vSphere Distributed Switches if you are not actively using it.

Network health check is disabled by default. Once enabled, the health check packets contain information about the host, switch, and port that an attacker can potentially use. Use network health check only for troubleshooting, and turn it off when troubleshooting is finished.

7 Protect virtual traffic against impersonation and interception Layer 2 attacks by configuring a security policy on port groups or ports.

The security policy on distributed port groups and ports includes the following options:

- MAC address changes (see MAC Address Changes)
- Promiscuous mode (see Promiscuous Mode Operation)
- Forged transmits (see Forged Transmits)

You can view and change the current settings by selecting **Manage Distributed Port Groups** from the right-button menu of the distributed switch and selecting **Security** in the wizard. See the *vSphere Networking* documentation.

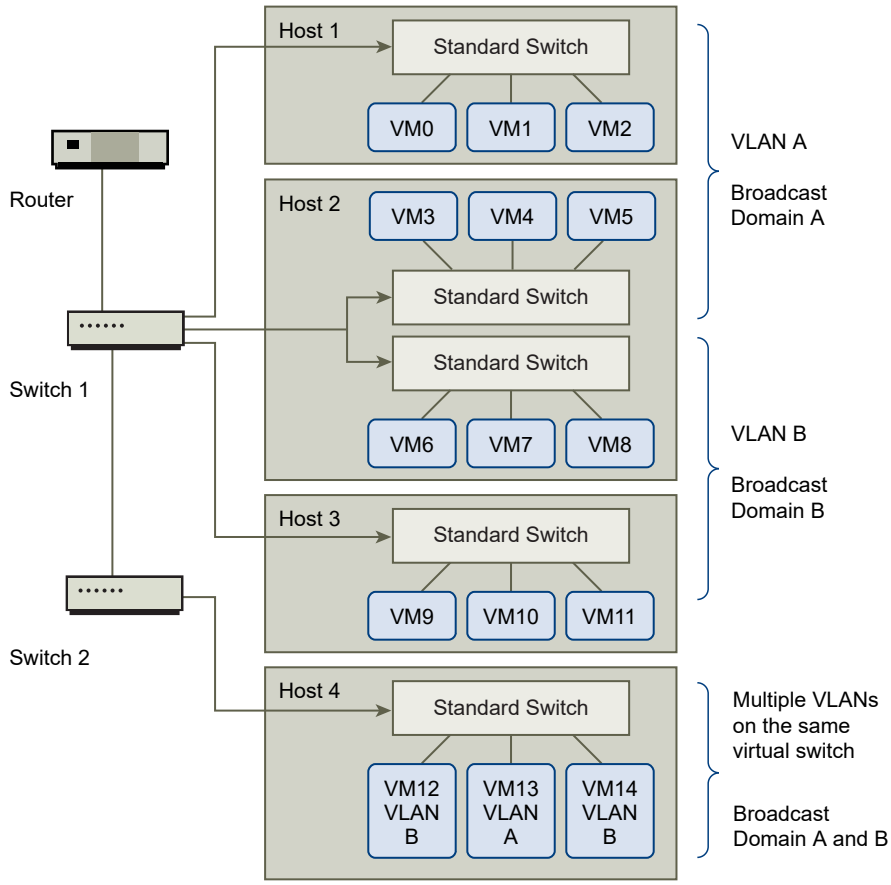
Securing Virtual Machines with VLANs

The network can be one of the most vulnerable parts of any system. Your virtual machine network requires as much protection as your physical network. Using VLANs can improve networking security in your environment.

VLANs are an IEEE standard networking scheme with specific tagging methods that allow routing of packets to only those ports that are part of the VLAN. When properly configured, VLANs provide a dependable means for you to protect a set of virtual machines from accidental or malicious intrusions.

VLANs let you segment a physical network so that two machines in the network are unable to transmit packets back and forth unless they are part of the same VLAN. For example, accounting records and transactions are among a company's most sensitive internal information. In a company whose sales, shipping, and accounting employees all use virtual machines in the same physical network, you might protect the virtual machines for the accounting department by setting up VLANs.

Figure 13-1. Sample VLAN Layout



In this configuration, all employees in the accounting department use virtual machines in VLAN A and the employees in sales use virtual machines in VLAN B.

The router forwards packets containing accounting data to the switches. These packets are tagged for distribution to VLAN A only. Therefore, the data is confined to Broadcast Domain A and cannot be routed to Broadcast Domain B unless the router is configured to do so.

This VLAN configuration prevents the sales force from intercepting packets destined for the accounting department. It also prevents the accounting department from receiving packets intended for the sales group. The virtual machines serviced by a single virtual switch can be in different VLANs.

Security Considerations for VLANs

The way you set up VLANs to secure parts of a network depends on factors such as the guest operating system and the way your network equipment is configured.

ESXi features a complete IEEE 802.1q-compliant VLAN implementation. VMware cannot make specific recommendations on how to set up VLANs, but there are factors to consider when using a VLAN deployment as part of your security enforcement policy.

Secure VLANs

Administrators have several options for securing the VLANs in their vSphere environment.

Procedure

1. Ensure that port groups are not configured to VLAN values that are reserved by upstream physical switches

   Do not set VLAN IDs to values reserved for the physical switch.

2. Ensure that port groups are not configured to VLAN 4095 unless you are using for Virtual Guest Tagging (VGT).

   Three types of VLAN tagging exist in vSphere:

   - External Switch Tagging (EST)
   - Virtual Switch Tagging (VST) - The virtual switch tags with the configured VLAN ID the traffic that is incoming to the attached virtual machines and removes the VLAN tag from the traffic that is leaving them. To set up VST mode, assign a VLAN ID between 1 and 4094.
   - Virtual Guest Tagging (VGT) - Virtual machines handle VLAN traffic. To activate VGT mode, set the VLAN ID to 4095. On a distributed switch, you can also allow virtual machine traffic based on its VLAN by using the **VLAN Trunking** option.

   On a standard switch you can configure VLAN networking mode at switch or port group level, and on a distributed switch at distributed port group or port level.

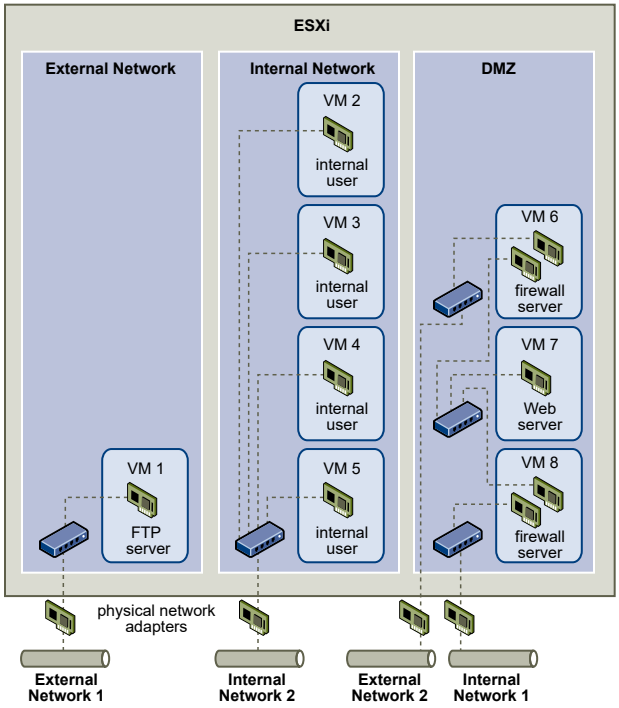
3. Ensure that all VLANs on each virtual switch are fully documented and that each virtual switch has all required VLANs and only required VLANs.

Creating Multiple Networks Within a Single ESXi Host

The ESXi system is designed so that you can connect some groups of virtual machines to the internal network, others to the external network, and still others to both—all on the same host.

This capability is an outgrowth of basic virtual machine isolation coupled with a well-planned use of virtual networking features.

Figure 13-2. External Networks, Internal Networks, and a DMZ Configured on a Single ESXi Host



In the figure, the system administrator configured a host into three distinct virtual machine zones: FTP server, internal virtual machines, and DMZ. Each zone serves a unique function.

FTP server

Virtual Machine 1 is configured with FTP software and acts as a holding area for data sent to and from outside resources such as forms and collateral localized by a vendor.

This virtual machine is associated with an external network only. It has its own virtual switch and physical network adapter that connect it to External Network 1. This network is dedicated to servers that the company uses to receive data from outside sources. For example, the company uses External Network 1 to receive FTP traffic from vendors and allow vendors access to data stored on externally available servers though FTP. In addition to servicing Virtual Machine 1, External Network 1 services FTP servers configured on different ESXi hosts throughout the site.

Because Virtual Machine 1 does not share a virtual switch or physical network adapter with any virtual machines in the host, the other resident virtual machines cannot transmit packets to or receive packets from the Virtual Machine 1 network. This restriction prevents sniffing attacks, which require sending network traffic to the victim. More importantly, an attacker cannot use the natural vulnerability of FTP to access any of the host's other virtual machines.

Internal virtual machines

Virtual Machines 2 through 5 are reserved for internal use. These virtual machines process and store company-private data such as medical records, legal settlements, and fraud investigations. As a result, the system administrators must ensure the highest level of protection for these virtual machines.

These virtual machines connect to Internal Network 2 through their own virtual switch and network adapter. Internal Network 2 is reserved for internal use by personnel such as claims processors, in-house lawyers, or adjustors.

Virtual Machines 2 through 5 can communicate with one another through the virtual switch and with internal virtual machines elsewhere on Internal Network 2 through the physical network adapter. They cannot communicate with externally facing machines. As with the FTP server, these virtual machines cannot send packets to or receive packets from the other virtual machines' networks. Similarly, the host's other virtual machines cannot send packets to or receive packets from Virtual Machines 2 through 5.

DMZ

Virtual Machines 6 through 8 are configured as a DMZ that the marketing group uses to publish the company's external website.

This group of virtual machines is associated with External Network 2 and Internal Network 1. The company uses External Network 2 to support the Web servers that use the marketing and financial department to host the corporate website and other Web facilities that it hosts to outside users. Internal Network 1 is the conduit that the marketing department uses to publish its content to the corporate Web site, post downloads, and maintain services like user forums.

Because these networks are separate from External Network 1 and Internal Network 2, and the virtual machines have no shared points of contact (switches or adapters), there is no risk of attack to or from the FTP server or the internal virtual machine group.

By capitalizing on virtual machine isolation, correctly configuring virtual switches, and maintaining network separation, the system administrator can house all three virtual machine zones in the same ESXi host and be confident that there will be no data or resource breaches.

The company enforces isolation among the virtual machine groups by using multiple internal and external networks and making sure that the virtual switches and physical network adapters for each group are separate from those of other groups.

Because none of the virtual switches straddle virtual machine zones, the system administrator succeeds in eliminating the risk of packet leakage from one zone to another. A virtual switch, by design, cannot leak packets directly to another virtual switch. The only way for packets to travel from one virtual switch to another is under the following circumstances:

- The virtual switches are connected to the same physical LAN.
- The virtual switches connect to a common virtual machine, which might be used to transmit packets.

Neither of these conditions occur in the sample configuration. If system administrators want to verify that no common virtual switch paths exist, they can check for possible shared points of contact by reviewing the network switch layout in the vSphere Client.

To safeguard the virtual machines' resources, the system administrator lowers the risk of DoS and DDoS attacks by configuring a resource reservation and a limit for each virtual machine. The system administrator further protects the ESXi host and virtual machines by installing software firewalls at the front and back ends of the DMZ, ensuring that the host is behind a physical firewall, and configuring the networked storage resources so that each has its own virtual switch.

Internet Protocol Security

Internet Protocol Security (IPsec) secures IP communications coming from and arriving at a host. ESXi hosts support IPsec using IPv6.

When you set up IPsec on a host, you enable authentication and encryption of incoming and outgoing packets. When and how IP traffic is encrypted depends on how you set up the system's security associations and security policies.

A security association determines how the system encrypts traffic. When you create a security association, you specify the source and destination, encryption parameters, and a name for the security association.

A security policy determines when the system should encrypt traffic. The security policy includes source and destination information, the protocol and direction of traffic to be encrypted, the mode (transport or tunnel) and the security association to use.

List Available Security Associations

ESXi can provide a list of all security associations available for use by security policies. The list includes both user created security associations and any security associations the VMkernel installed using Internet Key Exchange.

You can get a list of available security associations using the `esxcli` command.

Procedure

- At the command prompt, enter the command **`esxcli network ip ipsec sa list`**.

Results

ESXi displays a list of all available security associations.

Add an IPsec Security Association

Add a security association to specify encryption parameters for associated IP traffic.

You can add a security association using the `esxcli` command.

Procedure

- At the command prompt, enter the command **`esxcli network ip ipsec sa add`** with one or more of the following options.

Option	Description
`--sa-source=` *source address*	Required. Specify the source address.
`--sa-destination=` *destination address*	Required. Specify the destination address.
`--sa-mode=` *mode*	Required. Specify the mode, either `transport` or `tunnel`.
`--sa-spi=` *security parameter index*	Required. Specify the security parameter index. The security parameter index identifies the security association to the host. It must be a hexadecimal with a 0x prefix. Each security association you create must have a unique combination of protocol and security parameter index.
`--encryption-algorithm=` *encryption algorithm*	Required. Specify the encryption algorithm using one of the following parameters. ■ `3des-cbc` ■ `aes128-cbc` ■ `null` (provides no encryption)
`--encryption-key=` *encryption key*	Required when you specify an encryption algorithm. Specify the encryption key. You can enter keys as ASCII text or as a hexadecimal with a 0x prefix.
`--integrity-algorithm=` *authentication algorithm*	Required. Specify the authentication algorithm, either `hmac-sha1` or `hmac-sha2-256`.
`--integrity-key=` *authentication key*	Required. Specify the authentication key. You can enter keys as ASCII text or as a hexadecimal with a 0x prefix.
`--sa-name=`*name*	Required. Provide a name for the security association.

Example: New Security Association Command

The following example contains extra line breaks for readability.

```
esxcli network ip ipsec sa add
--sa-source 3ffe:501:ffff:0::a
--sa-destination 3ffe:501:ffff:0001:0000:0000:0000:0001
--sa-mode transport
--sa-spi 0x1000
--encryption-algorithm 3des-cbc
--encryption-key 0x697076367265616479 6c6f676f336465736362636f757432
--integrity-algorithm hmac-sha1
--integrity-key 0x69707636726561647 96c6f67736861316f757432
--sa-name sa1
```

Remove an IPsec Security Association

You can remove a security association using the ESXCLI command.

Prerequisites

Verify that the security association you want to use is not currently in use. If you try to remove a security association that is in use, the removal operation fails.

Procedure

- At the command prompt, enter the command **`esxcli network ip ipsec sa remove --sa-name`** ***`security_association_name`***.

List Available IPsec Security Policies

You can list available security policies using the ESXCLI command.

Procedure

- At the command prompt, enter the command **`esxcli network ip ipsec sp list`**.

Results

The host displays a list of all available security policies.

Create an IPSec Security Policy

Create a security policy to determine when to use the authentication and encryption parameters set in a security association. You can add a security policy using the ESXCLI command.

Prerequisites

Before creating a security policy, add a security association with the appropriate authentication and encryption parameters as described in Add an IPsec Security Association.

Procedure

- At the command prompt, enter the command **`esxcli network ip ipsec sp add`** with one or more of the following options.

Option	Description
`--sp-source=` ***`source address`***	Required. Specify the source IP address and prefix length.
`--sp-destination=` ***`destination address`***	Required. Specify the destination address and prefix length.
`--source-port=` ***`port`***	Required. Specify the source port. The source port must be a number between 0 and 65535.
`--destination-port=` ***`port`***	Required. Specify the destination port. The source port must be a number between 0 and 65535.
`--upper-layer-protocol=` ***`protocol`***	Specify the upper layer protocol using one of the following parameters. ■ `tcp` ■ `udp` ■ `icmp6` ■ `any`

Option	Description
--flow-direction= ***direction***	Specify the direction in which you want to monitor traffic using either `in` or `out`.
--action= ***action***	Specify the action to take when traffic with the specified parameters is encountered using one of the following parameters. ■ `none`: Take no action. ■ `discard`: Do not allow data in or out. ■ `ipsec`: Use the authentication and encryption information supplied in the security association to determine whether the data comes from a trusted source.
--sp-mode= ***mode***	Specify the mode, either `tunnel` or `transport`.
--sa-name=*security association name*	Required. Provide the name of the security association for the security policy to use.
--sp-name=*name*	Required. Provide a name for the security policy.

Example: New Security Policy Command

The following example includes extra line breaks for readability.

```
esxcli network ip ipsec add
--sp-source=2001:db8:1::/64
--sp-destination=2002:db8:1::/64
--source-port=23
--destination-port=25
--upper-layer-protocol=tcp
--flow-direction=out
--action=ipsec
--sp-mode=transport
--sa-name=sa1
--sp-name=sp1
```

Remove an IPsec Security Policy

You can remove a security policy from the ESXi host using the ESXCLI command.

Prerequisites

Verify that the security policy you want to use is not currently in use. If you try to remove a security policy that is in use, the removal operation fails.

Procedure

- At the command prompt, enter the command **esxcli network ip ipsec sp remove --sa-name *security policy name***.

  To remove all security policies, enter the command **esxcli network ip ipsec sp remove --remove-all**.

Ensure Proper SNMP Configuration

If SNMP is not properly configured, monitoring information can be sent to a malicious host. The malicious host can then use this information to plan an attack.

ESXi includes an SNMP agent that can send notifications (traps and informs) and receive `GET`, `GETBULK`, and `GETNEXT` requests. SNMP is not enabled by default. SNMP must be configured on each ESXi host. You can use ESXCLI, PowerCLI, or the vSphere Web Services SDK for configuration.

See the *vSphere Monitoring and Performance* documentation for detailed information about configuring SNMP, including SNMP v3. SNMP v3 provides stronger security than SNMP v1 or SNMP v2c, including key authentication and encryption. See *ESXCLI Reference* for more information about the `esxcli system snmp` command options.

Procedure

1. To determine whether SNMP is used, run the following command.

   ```
   esxcli system snmp get
   ```

2. To enable SNMP, run the following command.

   ```
   esxcli system snmp set --enable true
   ```

3. To disable SNMP, run the following command.

   ```
   esxcli system snmp set --enable false
   ```

vSphere Networking Security Best Practices

Following networking security best practices helps ensure the integrity of your vSphere deployment.

General Networking Security Recommendations

Following general network security recommendations is the first step in securing your networking environment. You can then move on to special areas, such as securing the network with firewalls or using IPsec.

- Spanning Tree Protocol (STP) detects and prevents loops from forming in the network topology. VMware virtual switches prevent loops in other ways, but do not support STP directly. When network topology changes occur, some time is required (30–50 seconds) while the network relearns the topology. During that time, no traffic is allowed to pass. To avoid these problems, network vendors have created features to enable switch ports to continue forwarding traffic. For more information, see the VMware knowledge base article at https://kb.vmware.com/kb/1003804. Consult your network vendor documentation for the proper network and networking hardware configurations.

- Ensure that Netflow traffic for a Distributed Virtual Switch is only sent to authorized collector IP addresses. Netflow exports are not encrypted and can contain information about the virtual network. This information increases the potential for sensitive information to be viewed and captured in transit by attackers. If Netflow export is required, verify that all Netflow target IP addresses are correct.
- Ensure that only authorized administrators have access to virtual networking components by using the role-based access controls. For example, give virtual machine administrators only access to port groups in which their virtual machines reside. Give network administrators access to all virtual networking components but no access to virtual machines. Limiting access reduces the risk of misconfiguration, whether accidental or malicious, and enforces key security concepts of separation of duties and least privilege.
- Ensure that port groups are not configured to the value of the native VLAN. Physical switches are often configured with a native VLAN, and that native VLAN is often VLAN 1 by default. ESXi does not have a native VLAN. Frames with VLAN specified in the port group have a tag, but frames with VLAN not specified in the port group are not tagged. This can cause a problem because virtual machines that are tagged with a 1 end up belonging to native VLAN of the physical switch.

  For example, frames on VLAN 1 from a Cisco physical switch are untagged because VLAN 1 is the native VLAN on that physical switch. However, frames from the ESXi host that are specified as VLAN 1 are tagged with a 1. As a result, traffic from the ESXi host that is destined for the native VLAN is not routed correctly because it is tagged with a 1 instead of being untagged. Traffic from the physical switch that is coming from the native VLAN is not visible because it is not tagged. If the ESXi virtual switch port group uses the native VLAN ID, traffic from virtual machines on that port is not visible to the native VLAN on the switch because the switch is expecting untagged traffic.
- Ensure that port groups are not configured to VLAN values reserved by upstream physical switches. Physical switches reserve certain VLAN IDs for internal purposes and often disallow traffic configured to these values. For example, Cisco Catalyst switches typically reserve VLANs 1001–1024 and 4094. Using a reserved VLAN might result in a denial of service on the network.
- Ensure that port groups are not configured to VLAN 4095 except for Virtual Guest Tagging (VGT). Setting a port group to VLAN 4095 activates VGT mode. In this mode, the virtual switch passes all network frames to the virtual machine without modifying the VLAN tags, leaving it to the virtual machine to deal with them.
- Restrict port-level configuration overrides on a distributed virtual switch. Port-level configuration overrides are disabled by default. When overrides are enabled, you can use different security settings for a virtual machine than the port-group level settings. Certain virtual machines require unique configurations, but monitoring is essential. If overrides are not monitored, anyone who gains access to a virtual machine with a less secure distributed virtual switch configuration might attempt to exploit that access.

- Ensure that distributed virtual switch port mirror traffic is sent only to authorized collector ports or VLANs. A vSphere Distributed Switch can mirror traffic from one port to another to allow packet capture devices to collect specific traffic flows. Port mirroring sends a copy of all specified traffic in unencrypted format. This mirrored traffic contains the full data in the packets captured and can result in total compromise of that data if misdirected. If port mirroring is required, verify that all port mirror destination VLAN, port, and uplink IDs are correct.

Labeling Networking Components

Identifying the different components of your networking architecture is critical and helps ensure that no errors are introduced as your network grows.

Follow these best practices:

- Ensure that port groups are configured with a clear network label. These labels serve as a functional descriptor for the port group and help you identify each port group's function as the network becomes more complex.
- Ensure that each vSphere Distributed Switch has a clear network label that indicates the function or IP subnet of the switch. This label serves as a functional descriptor for the switch, just as physical switches require a host name. For example, you can label the switch as internal to show that it is for internal networking. You cannot change the label for a standard virtual switch.

Document and Check the vSphere VLAN Environment

Check your VLAN environment regularly to avoid addressing problems. Fully document the VLAN environment and ensure that VLAN IDs are used only once. Your documentation can help with troubleshooting and is essential when you want to expand the environment.

Procedure

1. Ensure that all vSwitch and VLANS IDs are fully documented

   If you are using VLAN tagging on a virtual switch, the IDs must correspond to the IDs on external VLAN-aware upstream switches. If VLAN IDs are not tracked completely, mistaken reuse of IDs might allow for traffic between the wrong physical and virtual machines. Similarly, if VLAN IDs are wrong or missing, traffic between physical and virtual machines might be blocked where you want traffic to pass.

2. Ensure that VLAN IDs for all distributed virtual port groups (dvPortgroup instances) are fully documented.

   If you are using VLAN tagging on a dvPortgroup the IDs must correspond to the IDs on external VLAN-aware upstream switches. If VLAN IDs are not tracked completely, mistaken reuse of IDs might allow for traffic between the wrong physical and virtual machines. Similarly, if VLAN IDs are wrong or missing, traffic between physical and virtual machines might be blocked where you want traffic to pass.

3 Ensure that private VLAN IDs for all distributed virtual switches are fully documented.

   Private VLANs (PVLANs) for distributed virtual switches require primary and secondary VLAN IDs. These IDs must correspond to the IDs on external PVLAN-aware upstream switches. If VLAN IDs are not tracked completely, mistaken reuse of IDs might allow for traffic between the wrong physical and virtual machines. Similarly, if PVLAN IDs are wrong or missing, traffic between physical and virtual machines might be blocked where you want traffic to pass.

4 Verify that VLAN trunk links are connected only to physical switch ports that function as trunk links.

   When connecting a virtual switch to a VLAN trunk port, you must properly configure both the virtual switch and the physical switch at the uplink port. If the physical switch is not properly configured, frames with the VLAN 802.1q header are forwarded to a switch that not expecting their arrival.

Adopting Network Isolation Practices

Network isolation practices bolster network security in your vSphere environment.

Isolate the Management Network

The vSphere management network provides access to the vSphere management interface on each component. Services running on the management interface provide an opportunity for an attacker to gain privileged access to the systems. Remote attacks are likely to begin with gaining access to this network. If an attacker gains access to the management network, it provides the staging ground for further intrusion.

Strictly control access to management network by protecting it at the security level of the most secure VM running on an ESXi host or cluster. No matter how the management network is restricted, administrators must have access to this network to configure the ESXi hosts and vCenter Server system.

Place the vSphere management port group in a dedicated VLAN on a common standard switch. Production (VM) traffic can share the standard switch if the vSphere management port group's VLAN is not used by production VMs.

Check that the network segment is not routed, except to networks where other management-related entities are found. Routing a network segment might make sense for vSphere Replication. In particular, make sure that production VM traffic cannot be routed to this network.

Strictly control access to management functionality by using one of the following approaches.

- To access the management network in especially sensitive environments, configure a controlled gateway or other controlled method. For example, require that administrators connect to the management network through a VPN. Allow access to the management network only to trusted administrators.
- Configure bastion hosts that run management clients.

Isolate Storage Traffic

Ensure that IP-based storage traffic is isolated. IP-based storage includes iSCSI and NFS. VMs might share virtual switches and VLANs with the IP-based storage configurations. This type of configuration might expose IP-based storage traffic to unauthorized VM users.

IP-based storage frequently is not encrypted. Anyone with access to this network can view IP-based storage traffic. To restrict unauthorized users from viewing IP-based storage traffic, logically separate the IP-based storage network traffic from the production traffic. Configure the IP-based storage adapters on separate VLANs or network segments from the VMkernel management network to limit unauthorized users from viewing the traffic.

Isolate vMotion Traffic

vMotion migration information is transmitted in plain text. Anyone with access to the network over which this information flows can view it. Potential attackers can intercept vMotion traffic to obtain the memory contents of a VM. They might also stage a MiTM attack in which the contents are modified during migration.

Separate vMotion traffic from production traffic on an isolated network. Set up the network to be nonroutable, that is, make sure that no layer-3 router is spanning this and other networks, to prevent outside access to the network.

Use a dedicated VLAN on a common standard switch for the vMotion port group. Production (VM) traffic can use the same standard switch if the vMotion port group's VLAN is not used by production VMs.

Isolate vSAN Traffic

When configuring your vSAN network, isolate vSAN traffic on its own Layer 2 network segment. You can perform this isolation by using dedicated switches or ports, or by using a VLAN.

Use Virtual Switches with the vSphere Network Appliance API Only If Required

Do not configure your host to send network information to a virtual machine unless you are using products that use the vSphere Network Appliance API (DvFilter). If the vSphere Network Appliance API is enabled, an attacker might attempt to connect a virtual machine to the filter. This connection might provide access to the network of other virtual machines on the host.

If you are using a product that uses this API, verify that the host is configured correctly. See the sections on DvFilter in *Developing and Deploying vSphere Solutions, vServices, and ESX Agents*. If your host is set up to use the API, make sure that the value of the `Net.DVFilterBindIpAddress` parameter matches the product that uses the API.

Procedure

1. Browse to the host in the vSphere Client inventory.
2. Click **Configure**.

3 Under System, click **Advanced System Settings**.

4 Scroll down to `Net.DVFilterBindIpAddress` and verify that the parameter has an empty value.

The order of parameters is not strictly alphabetical. Type **`DVFilter`** in the Filter text box to display all related parameters.

5 Verify the setting.

- If you are not using DvFilter settings, make sure that the value is blank.
- If you are using DvFilter settings, make sure that the value of the parameter is correct. The value must match the value that the product that uses the DvFilter is using.

Best Practices Involving Multiple vSphere Components

14

Some security best practices, such as setting up PTP or NTP in your environment, affect more than one vSphere component. Consider these recommendations when configuring your environment.

See Chapter 3 Securing ESXi Hosts and Chapter 5 Securing Virtual Machines for related information.

Read the following topics next:

- Synchronizing Clocks on the vSphere Network
- Storage Security Best Practices
- Verify That Sending Host Performance Data to Guests Is Disabled
- Setting Timeouts for the ESXi Shell and vSphere Client

Synchronizing Clocks on the vSphere Network

Verify that all components on the vSphere network have their clocks synchronized. If the clocks on the physical machines in your vSphere network are not synchronized, SSL certificates and SAML Tokens, which are time-sensitive, might not be recognized as valid in communications between network machines.

Unsynchronized clocks can result in authentication problems, which can cause the installation to fail or prevent the vCenter Server `vmware-vpxd` service from starting.

Time inconsistencies in vSphere can cause firstboot to fail at different services depending on where in the environment time is not accurate and when the time is synchronized. Problems most commonly occur when the target ESXi host for the destination vCenter Server is not synchronized with NTP or PTP. Similarly, issues can arise if the destination vCenter Server migrates to an ESXi host set to a different time due to fully automated DRS.

To avoid time synchronization issues, ensure that the following is correct before installing, migrating, or upgrading a vCenter Server.

- The target ESXi host where the destination vCenter Server is to be deployed is synchronized to NTP or PTP.
- The ESXi host running the source vCenter Server is synchronized to NTP or PTP.

- When upgrading or migrating from vSphere 6.5 or 6.7 to vSphere 7.0, if the vCenter Server appliance is connected to an external Platform Services Controller, ensure the ESXi host running the external Platform Services Controller is synchronized to NTP or PTP.
- If you are upgrading or migrating from vSphere 6.5 or 6.7 to vSphere 7.0, verify that the source vCenter Server or vCenter Server appliance and external Platform Services Controller have the correct time.
- When you upgrade a vCenter Server 6.5 or 6.7 instance with an external Platform Services Controller to vSphere 7.0, the upgrade process converts to a vCenter Server instance with an embedded Platform Services Controller.

Verify that any Windows host machine on which vCenter Server runs is synchronized with the Network Time Server (NTP) server. See the VMware knowledge base article at https://kb.vmware.com/s/article/1318.

To synchronize ESXi clocks with an NTP server or a PTP server, you can use the VMware Host Client. For information about editing the time configuration of an ESXi host, see *vSphere Single Host Management - VMware Host Client*.

To learn how to change time synchronization settings for vCenter Server, see "Configure the System Time Zone and Time Synchronization Settings" in *vCenter Server Configuration*.

To learn how to edit time configuration for a host by using the vSphere Client, see "Editing Time Configuration for a Host" in *vCenter Server and Host Management*.

What to read next

- Synchronize ESXi Clocks with a Network Time Server

  Before you install vCenter Server, make sure all machines on your vSphere network have their clocks synchronized.
- Configuring Time Synchronization Settings in vCenter Server

  You can change the time synchronization settings in vCenter Server after deployment.

Synchronize ESXi Clocks with a Network Time Server

Before you install vCenter Server, make sure all machines on your vSphere network have their clocks synchronized.

This task explains how to set up NTP from the VMware Host Client.

Procedure

1. Start the VMware Host Client, and connect to the ESXi host.
2. Click **Manage**.
3. Under **System**, click **Time & date**, and click **Edit settings**.
4. Select **Use Network Time Protocol (enable NTP client)**.

5 In the NTP servers text box, enter the IP address or fully qualified domain name of one or more NTP servers to synchronize with.

6 From the **NTP Service Start-up Policy** drop-down menu, select **Start and stop with host**.

7 Click **Save**.

The host synchronizes with the NTP server.

Configuring Time Synchronization Settings in vCenter Server

You can change the time synchronization settings in vCenter Server after deployment.

When you deploy vCenter Server, you can choose the time synchronization method to be either by using an NTP server or by using VMware Tools. In case the time settings in your vSphere network change, you can edit the vCenter Server and configure the time synchronization settings by using the commands in the appliance shell.

When you enable periodic time synchronization, VMware Tools sets the time of the guest operating system to be the same as the time of the host.

After time synchronization occurs, VMware Tools checks once every minute to determine whether the clocks on the guest operating system and the host still match. If not, the clock on the guest operating system is synchronized to match the clock on the host.

Native time synchronization software, such as Network Time Protocol (NTP), is typically more accurate than VMware Tools periodic time synchronization and is therefore preferred. You can use only one form of periodic time synchronization in vCenter Server. If you decide to use native time synchronization software, vCenter Server VMware Tools periodic time synchronization is disabled, and the reverse.

Use VMware Tools Time Synchronization

You can set up vCenter Server to use VMware Tools time synchronization.

Procedure

1 Access the appliance shell and log in as a user who has the administrator or super administrator role.

The default user with super administrator role is root.

2 Run the command to enable VMware Tools time synchronization.

```
timesync.set --mode host
```

3 (Optional) Run the command to verify that you successfully applied the VMware Tools time synchronization.

```
timesync.get
```

The command returns that the time synchronization is in host mode.

Results

The time of the appliance is synchronized with the time of the ESXi host.

Add or Replace NTP Servers in the vCenter Server Configuration

To set up the vCenter Server to use NTP-based time synchronization, you must add the NTP servers to the vCenter Server configuration.

Procedure

1. Access the appliance shell and log in as a user who has the administrator or super administrator role.

   The default user with super administrator role is root.

2. Add NTP servers to the vCenter Server configuration by running the following `ntp.set` command.

   ```
   ntp.set --servers IP-addresses-or-host-names
   ```

   In this command, *IP-addresses-or-host-names* is a comma-separated list of IP addresses or host names of the NTP servers.

   This command removes the current NTP servers (if any) and adds the new NTP servers to the configuration. If the time synchronization is based on an NTP server, then the NTP daemon is restarted to reload the new NTP servers. Otherwise, this command replaces the current NTP servers in the NTP configuration with the new NTP servers you specify.

3. (Optional) To verify that you successfully applied the new NTP configuration settings, run the following command.

   ```
   ntp.get
   ```

   The command returns a space-separated list of the servers configured for NTP synchronization. If the NTP synchronization is enabled, the command returns that the NTP configuration is in Up status. If the NTP synchronization is disabled, the command returns that the NTP configuration is in Down status.

4. (Optional) To verify if the NTP server is reachable, run the following command.

   ```
   ntp.test --servers IP-addresses-or-host-names
   ```

   The command returns the status of the NTP servers.

What to do next

If the NTP synchronization is disabled, you can configure the time synchronization settings in the vCenter Server to be based on an NTP server. See Synchronize the Time in vCenter Server with an NTP Server.

Synchronize the Time in vCenter Server with an NTP Server

You can configure the time synchronization settings in the vCenter Server to be based on an NTP server.

Prerequisites

Set up one or more Network Time Protocol (NTP) servers in the vCenter Server configuration. See Add or Replace NTP Servers in the vCenter Server Configuration.

Procedure

1. Access the appliance shell and log in as a user who has the administrator or super administrator role.

   The default user with super administrator role is root.

2. Run the command to enable NTP-based time synchronization.

   ```
   timesync.set --mode NTP
   ```

3. (Optional) Run the command to verify that you successfully applied the NTP synchronization.

   ```
   timesync.get
   ```

   The command returns that the time synchronization is in NTP mode.

Storage Security Best Practices

Follow best practices for storage security, as outlined by your storage security provider. You can also take advantage of CHAP and mutual CHAP to secure iSCSI storage, mask and zone SAN resources, and configure Kerberos credentials for NFS 4.1.

See also the *Administering VMware vSAN* documentation.

Securing iSCSI Storage

The storage you configure for a host might include one or more storage area networks (SANs) that use iSCSI. When you configure iSCSI on a host, you can take measures to minimize security risks.

iSCSI supports accessing SCSI devices and exchanging data by using TCP/IP over a network port rather than through a direct connection to a SCSI device. An iSCSI transaction encapsulates blocks of raw SCSI data in iSCSI records and transmits the data to the requesting device or user.

iSCSI SANs support efficient use of the existing Ethernet infrastructure to provide hosts access to storage resources that they can dynamically share. iSCSI SANs are an economical storage solution for environments that rely on a common storage pool to serve many users. As with any networked system, your iSCSI SANs can be subject to security breaches.

Note The requirements and procedures for securing an iSCSI SAN are similar for hardware iSCSI adapters associated with hosts and for iSCSI configured directly through the host.

Securing iSCSI Devices

To secure iSCSI devices, require that the ESXi host, or initiator, can authenticate to the iSCSI device, or target, whenever the host attempts to access data on the target LUN.

Authentication ensures that the initiator has the right to access a target. You grant this right when you configure authentication on the iSCSI device.

ESXi does not support Secure Remote Protocol (SRP), or public-key authentication methods for iSCSI. You can use Kerberos only with NFS 4.1.

ESXi supports both CHAP and Mutual CHAP authentication. The *vSphere Storage* documentation explains how to select the best authentication method for your iSCSI device and how to set up CHAP.

Ensure uniqueness of CHAP secrets. Set up a different mutual authentication secret for each host. If possible, set up a different secret for each client that to the ESXi host. Unique secrets ensure that an attacker cannot create another arbitrary host and authenticate to the storage device even if one host is compromised. With a shared secret, compromise of one host might allow an attacker to authenticate to the storage device.

Protecting an iSCSI SAN

When you plan your iSCSI configuration, take measures to improve the overall security of the iSCSI SAN. Your iSCSI configuration is only as secure as your IP network, so by enforcing good security standards when you set up your network, you help safeguard your iSCSI storage.

The following are some specific suggestions for enforcing good security standards.

Protect Transmitted Data

A primary security risk in iSCSI SANs is that an attacker might sniff transmitted storage data.

Take additional measures to prevent attackers from easily seeing iSCSI data. Neither the hardware iSCSI adapter nor ESXi iSCSI initiator encrypts the data that they transmit to and from the targets, making the data more vulnerable to sniffing attacks.

Allowing your virtual machines to share standard switches and VLANs with your iSCSI configuration potentially exposes iSCSI traffic to misuse by a virtual machine attacker. To help ensure that intruders cannot listen to iSCSI transmissions, make sure that none of your virtual machines can see the iSCSI storage network.

If you use a hardware iSCSI adapter, you can accomplish this by making sure that the iSCSI adapter and ESXi physical network adapter are not inadvertently connected outside the host by virtue of sharing a switch or some other means. If you configure iSCSI directly through the ESXi host, you can accomplish this by configuring iSCSI storage through a different standard switch than the one used by your virtual machines.

In addition to protecting the iSCSI SAN by giving it a dedicated standard switch, you can configure your iSCSI SAN on its own VLAN to improve performance and security. Placing your iSCSI configuration on a separate VLAN ensures that no devices other than the iSCSI adapter have visibility into transmissions within the iSCSI SAN. Also, network congestion from other sources cannot interfere with iSCSI traffic.

Secure iSCSI Ports

When you run iSCSI devices, ESXi does not open any ports that listen for network connections. This measure reduces the chances that an intruder can break into ESXi through spare ports and gain control over the host. Therefore, running iSCSI does not present any additional security risks at the ESXi end of the connection.

Any iSCSI target device that you run must have one or more open TCP ports to listen for iSCSI connections. If any security vulnerabilities exist in the iSCSI device software, your data can be at risk through no fault of ESXi. To lower this risk, install all security patches that your storage equipment manufacturer provides and limit the devices connected to the iSCSI network.

Masking and Zoning SAN Resources

You can use zoning and LUN masking to separate SAN activity and restrict access to storage devices.

You can protect access to storage in your vSphere environment by using zoning and LUN masking with your SAN resources. For example, you might manage zones defined for testing independently within the SAN so they do not interfere with activity in the production zones. Similarly, you might set up different zones for different departments.

When you set up zones, take into account any host groups that are set up on the SAN device.

Zoning and masking capabilities for each SAN switch and disk array and the tools for managing LUN masking are vendor specific.

See your SAN vendor's documentation and the *vSphere Storage* documentation.

Using Kerberos for NFS 4.1

With NFS version 4.1, ESXi supports the Kerberos authentication mechanism.

The RPCSEC_GSS Kerberos mechanism is an authentication service. It allows an NFS 4.1 client installed on ESXi to prove its identity to an NFS server before mounting an NFS share. The Kerberos security uses cryptography to work across an insecure network connection.

The ESXi implementation of Kerberos for NFS 4.1 provides two security models, krb5 and krb5i, that offer different levels of security.

- Kerberos for authentication only (krb5) supports identity verification.
- Kerberos for authentication and data integrity (krb5i), in addition to identity verification, provides data integrity services. These services help to protect the NFS traffic from tampering by checking data packets for any potential modifications.

Kerberos supports cryptographic algorithms that prevent unauthorized users from gaining access to NFS traffic. The NFS 4.1 client on ESXi attempts to use either the AES256-CTS-HMAC-SHA1-96 or AES128-CTS-HMAC-SHA1-96 algorithm to access a share on the NAS server. Before using your NFS 4.1 datastores, make sure that AES256-CTS-HMAC-SHA1-96 or AES128-CTS-HMAC-SHA1-96 are enabled on the NAS server.

The following table compares Kerberos security levels that ESXi supports.

Table 14-1. Types of Kerberos Security

		ESXi 6.0	ESXi 6.5 and later
Kerberos for authentication only (krb5)	Integrity checksum for RPC header	Yes with DES	Yes with AES
	Integrity checksum for RPC data	No	No
Kerberos for authentication and data integrity (krb5i)	Integrity checksum for RPC header	No krb5i	Yes with AES
	Integrity checksum for RPC data		Yes with AES

When you use Kerberos authentication, the following considerations apply:

- ESXi uses Kerberos with the Active Directory domain.
- As a vSphere administrator, you specify Active Directory credentials to provide access to NFS 4.1 Kerberos datastores for an NFS user. A single set of credentials is used to access all Kerberos datastores mounted on that host.
- When multiple ESXi hosts share the NFS 4.1 datastore, you must use the same Active Directory credentials for all hosts that access the shared datastore. To automate the assignment process, set the user in host profiles and apply the profile to all ESXi hosts.
- You cannot use two security mechanisms, AUTH_SYS and Kerberos, for the same NFS 4.1 datastore shared by multiple hosts.

See the *vSphere Storage* documentation for step-by-step instructions.

Verify That Sending Host Performance Data to Guests Is Disabled

vSphere includes virtual machine performance counters on Windows operating systems where VMware Tools is installed. Performance counters allow virtual machine owners to do accurate performance analysis within the guest operating system. By default, vSphere does not expose host information to the guest virtual machine.

By default, the capability to send host performance data to a virtual machine is disabled. This default setting prevents a virtual machine from obtaining detailed information about the physical host. If a security breach of the virtual machine occurs, the setting does not make host data available to the attacker.

Note The following procedure illustrates the basic process. Consider using ESXCLI or VMware PowerCLI commands for performing this task on all hosts simultaneously.

Procedure

1. On the ESXi system that hosts the virtual machine, browse to the VMX file.

   Virtual machine configuration files are located in the `/vmfs/volumes/`*`datastore`* directory, where *datastore* is the name of the storage device where the virtual machine files are stored.

2. In the VMX file, verify that the following parameter is set.

   ```
   tools.guestlib.enableHostInfo=FALSE
   ```

3. Save and close the file.

Results

You cannot retrieve performance information about the host from inside the guest virtual machine.

Setting Timeouts for the ESXi Shell and vSphere Client

To prevent intruders from using an idle session, set timeouts for the ESXi Shell and vSphere Client.

ESXi Shell Timeout

For the ESXi Shell, you can set the following timeouts from the vSphere Client and from the Direct Console User Interface (DCUI).

Availability Timeout

The availability timeout setting is the amount of time that can elapse before you must log in after the ESXi Shell is enabled. After the timeout period, the service is disabled and users are not allowed to log in.

Idle Timeout

The idle timeout is the amount of time that can elapse before the user is logged out of an idle interactive sessions. Changes to the idle timeout apply the next time a user logs in to the ESXi Shell. Changes do not affect existing sessions.

vSphere Client Timeout

vSphere Client sessions end after 120 minutes by default. To change the default:

1 In the vSphere Client, navigate to the vCenter Server instance.

2 Select the **Configure** tab, and under **Settings**, select **General**.

3 Click **Edit**.

4 Select **Timeout settings**.

5 Enter your choices and click **Save**.

Managing TLS Protocol Configuration with the TLS Configurator Utility

15

vSphere enables only TLS by default. TLS 1.0 and TLS 1.1 are disabled by default. Whether you do a fresh install, upgrade, or migration, vSphere disables TLS 1.0 and TLS 1.1. You can use the TLS Configurator utility to enable older versions of the protocol temporarily on vSphere systems. You can then disable the older less secure versions after all connections use TLS 1.2.

Before you perform a reconfiguration, consider your environment. Depending on your environmental requirements and software versions, you might need to re-enable TLS 1.0 and TLS 1.1, in addition to TLS 1.2, to maintain interoperability. For VMware products, see the VMware Knowledge Base article at https://kb.vmware.com/s/article/2145796 for a list of VMware products that support TLS 1.2. For third-party integration, consult your vendor's documentation. The TLS Configurator utility works with vSphere 7.0 and prior releases, including 6.7, 6.5, and 6.0.

Read the following topics next:

Ports That Support Disabling TLS Versions

When you run the TLS Configurator utility in the vSphere environment, you can disable TLS across ports that use TLS on vCenter Server and ESXi hosts. You can disable TLS 1.0 or both TLS 1.0 and TLS 1.1.

Starting in vSphere 7.0, vCenter Server runs two reverse proxy services:

- VMware reverse proxy service, `rhttpproxy`
- Envoy

Envoy is an open source edge and service proxy. Envoy owns port 443, and all incoming vCenter Server requests are routed through Envoy. In vSphere 7.0, `rhttpproxy` serves as a configuration management server for Envoy. As a result, the TLS configuration is applied to `rhttpproxy`, which in turn sends the configuration to Envoy.

vCenter Server and ESXi use ports that can be enabled or disabled for TLS protocols. The TLS Configuration utility `scan` option displays which versions of TLS are enabled for each service. See Scan vCenter Server for Enabled TLS Protocols.

For the list of all supported ports and protocols in VMware products, including vSphere and vSAN, see the VMware Ports and Protocols Tool™ at https://ports.vmware.com/. You can search ports by VMware product, create a customized list of ports, and print or save port lists.

Notes and Caveats

- The vSphere 6.7 release was the final release of vCenter Server for Windows. See the *vSphere Security* documentation for the 6.7 version of the product for information about reconfiguring TLS for Update Manager ports on vCenter Server for Windows.
- You can use TLS 1.2 to encrypt the connection between vCenter Server and an external Microsoft SQL Server. You cannot use a TLS 1.2 only connection to an external Oracle database. See the VMware Knowledge Base article at https://kb.vmware.com/kb/2149745.
- For vSphere 6.7 and earlier releases, do not disable TLS 1.0 on a vCenter Server or Platform Services Controller instance that is running on Windows Server 2008. Windows 2008 supports only TLS 1.0. See the Microsoft TechNet Article *TLS/SSL Settings* in the *Server Roles and Technologies Guide*.
- If you change the TLS protocols, you must restart the ESXi host to apply the changes. You must restart the host even if you apply the changes through the cluster configuration by using host profiles. You can choose to restart the host immediately, or postpone the restart to a more convenient time.

Enabling or Disabling TLS Versions in vSphere

Disabling TLS versions is a multi-phase process. Disabling TLS versions in the right order ensures that your environment stays up and running during the process.

vSphere Lifecycle Manager is always included with the vCenter Server system and the script updates the corresponding port.

1 Run the TLS Configurator utility on vCenter Server.

2 Run the TLS Configurator utility on each ESXi host that is managed by the vCenter Server. You can perform this task for each host or for all hosts in a cluster.

Prerequisites

You have two choices for using TLS in your environment.

- Disable TLS 1.0, and enable TLS 1.1 and TLS 1.2.

- Disable TLS 1.0 and TLS 1.1, and enable TLS 1.2.

Perform an Optional Manual Backup

The TLS Configuration utility performs a backup each time the script modifies vCenter Server. If you need a backup to a specific directory, you can perform a manual backup.

Backup of the ESXi configuration is not supported.

For vCenter Server, the default directory is `/tmp/yearmonthdayTtime`.

Procedure

1. Change directory to `/usr/lib/vmware-TlsReconfigurator/VcTlsReconfigurator`.
2. To make a backup to a specific directory, run the following command.

   ```
   directory_path/VcTlsReconfigurator> ./reconfigureVc backup -d backup_directory_path
   ```

3. Verify that the backup was successful.

   A successful backup looks similar to the following example. The order of services displayed might be different each time you run the `reconfigureVc backup` command, due to the way the command runs.

   ```
   vCenter Transport Layer Security reconfigurator, version=7.0.0, build=15518531
   For more information refer to the following article: https://kb.vmware.com/kb/2147469
   Log file: "/var/log/vmware/vSphere-TlsReconfigurator/VcTlsReconfigurator.log".
   ================= Backing up vCenter Server TLS configuration ==================
   Using backup directory: /tmp/20200206T183550
   Backing up: vmware-rbd-watchdog
   Backing up: vmware-vpxd
   Backing up: vmcam
   Backing up: vmware-stsd
   Backing up: vmdird
   Backing up: vmware-sps
   Backing up: vmware-rhttpproxy
   Backing up: vami-lighttp
   Backing up: vmware-updatemgr
   Backing up: rsyslog
   ```

4. (Optional) If you later have to perform a restore, you can run the following command.

   ```
   reconfigureVc restore -d optional_custom_backup_directory_path
   ```

Enable or Disable TLS Versions on vCenter Server Systems

You can use the TLS Configuration utility to enable or disable TLS versions on vCenter Server systems. As part of the process, you can disable TLS 1.0, and enable TLS 1.1 and TLS 1.2. Or, you can disable TLS 1.0 and TLS 1.1, and enable only TLS 1.2.

Prerequisites

Ensure that the hosts and services that the vCenter Server manages can communicate using a version of TLS that remains enabled. For products that communicate only using TLS 1.0, connectivity becomes unavailable.

Procedure

1. Log in to the vCenter Server system with the user name and password for administrator@vsphere.local, or as another member of the vCenter Single Sign-On Administrators group who can run scripts.
2. Go to the directory where the script is located.

   ```
   cd /usr/lib/vmware-TlsReconfigurator/VcTlsReconfigurator
   ```

3. Run the command, depending on which version of TLS you want to use.

   Note The following commands restart the vCenter Server services.

   - To disable TLS 1.0 and enable both TLS 1.1 and TLS 1.2, run the following command.

     ```
     directory_path/VcTlsReconfigurator> ./reconfigureVc update -p TLSv1.1 TLSv1.2
     ```

   - To disable TLS 1.0 and TLS 1.1, and enable only TLS 1.2, run the following command.

     ```
     directory_path/VcTlsReconfigurator> ./reconfigureVc update -p TLSv1.2
     ```

4. If your environment includes other vCenter Server systems, repeat the process on each vCenter Server system.
5. Repeat the configuration on each ESXi host.

Enable or Disable TLS Versions on ESXi Hosts

You can use the TLS Configuration utility to enable or disable TLS versions on an ESXi host. As part of the process, you can disable TLS 1.0, and enable TLS 1.1 and TLS 1.2. Or, you can disable TLS 1.0 and TLS 1.1, and enable only TLS 1.2.

For ESXi hosts, you use a different utility than for the other components of your vSphere environment. The utility is release-specific, and cannot be used on a previous release.

You can write a script to configure multiple hosts.

Prerequisites

Ensure that any products or services associated with the ESXi host can communicate using TLS 1.1 or TLS 1.2. For products that communicate only using TLS 1.0, connectivity is lost.

The Bash shell must be enabled on the vCenter Server Appliance.

Procedure

1 Using SSH, connect to the vCenter Server Appliance with the user name and password of the vCenter Single Sign-On user who can run scripts.

2 To enable the Bash shell, enter **`shell`** at the command line.

3 Go to the directory where the script is located.

```
cd /usr/lib/vmware-TlsReconfigurator/EsxTlsReconfigurator
```

4 For an ESXi host that is part of a cluster, run one of the following commands.

- To disable TLS 1.0 and enable both TLS 1.1 and TLS 1.2 on all hosts in a cluster, run the following command.

```
./reconfigureEsx vCenterCluster -c Cluster_Name -u Administrative_User -p TLSv1.1
TLSv1.2
```

- To disable TLS 1.0 and TLS 1.1, and enable only TLS 1.2 on all hosts in a cluster, run the following command.

```
./reconfigureEsx vCenterCluster -c Cluster_Name -u Administrative_User -p TLSv1.2
```

5 For an individual host that is not part of a cluster, run one of the following commands.

- To disable TLS 1.0 and enable both TLS 1.1 and TLS 1.2 for an individual host, run the following command.

```
./reconfigureEsx vCenterHost -h ESXi_Host_Name -u Administrative_User -p TLSv1.1
TLSv1.2
```

- To disable TLS 1.0 and TLS 1.1, and enable only TLS 1.2 for an individual host, run the following command.

```
./reconfigureEsx vCenterHost -h ESXi_Host_Name -u Administrative_User -p TLSv1.2
```

Note To reconfigure a standalone ESXi host, log into a vCenter Server system and run the `reconfigureEsx` command with the `ESXiHost -h` *HOST* `-u` *ESXi_USER* options. For the *HOST* option, you can specify the IP address or FQDN of a single ESXi host, or a list of host IP addresses or FQDNs. For example, logging in to a vCenter Server and running the following command enables both TLS 1.1 and TLS 1.2 on two ESXi hosts:

```
./reconfigureEsx ESXiHost -h 198.51.100.2 198.51.100.3 -u root -p TLSv1.1 TLSv1.2
```

Alternatively, to reconfigure a standalone ESXi host, you can log into the host and modify the `UserVars.ESXiVPsDisabledProtocols` advanced setting. See the topic titled "Configure Advanced TLS/SSL Key Options" in the *vSphere Single Host Management - VMware Host Client* documentation for more information.

6 Reboot the ESXi host to complete the TLS protocol changes.

Scan vCenter Server for Enabled TLS Protocols

After you enable or disable TLS versions on vCenter Server, you can use the TLS Configuration utility to view your changes.

The TLS Configuration utility `scan` option displays which versions of TLS are enabled for each service.

Procedure

1. Log in to the vCenter Server system.
   a. Connect to the appliance using SSH and log in as a user who has privileges to run scripts.
   b. If the bash shell is not currently enabled, run the following commands.

```
shell.set --enabled true
shell
```

2. Go to the `VcTlsReconfigurator` directory.

```
cd /usr/lib/vmware-TlsReconfigurator/VcTlsReconfigurator
```

3. To display which services have TLS enabled, and the ports used, run the following command.

```
./reconfigureVc scan
```

Revert TLS Configuration Changes

You can use the TLS Configuration utility to revert configuration changes. When you revert the changes, the system enables protocols that you disabled using TLS Configurator utility.

Prerequisites

Before reverting changes, use the vCenter Server Management interface to perform a backup of the vCenter Server.

Procedure

1. Connect to the vCenter Server where you want to revert changes as a user who has privileges to run scripts.
2. If the Bash shell is not currently enabled, run the following commands.

```
shell.set --enabled true
shell
```

3. Go to the `VcTlsReconfigurator` directory.

```
cd /usr/lib/vmware-TlsReconfigurator/VcTlsReconfigurator
```

4 Review the previous backup.

```
grep "backup directory" /var/log/vmware/vSphere-TlsReconfigurator/VcTlsReconfigurator.log
```

The output looks like the following example.

```
2016-11-17T17:29:20.950Z INFO Using backup directory: /tmp/20161117T172920
2016-11-17T17:32:59.019Z INFO Using backup directory: /tmp/20161117T173259
```

5 Run the following command to perform a restore.

```
reconfigureVc restore -d Directory_path_from_previous_step
```

The output looks like the following example.

```
2016-11-17T17:29:20.950Z INFO Using backup directory: /tmp/20161117T172920
2016-11-17T17:32:59.019Z INFO Using backup directory: /tmp/20161117T173259
```

6 Repeat the procedure on any other vCenter Server instances.

Defined Privileges

16

The following tables list the default privileges that, when selected for a role, can be paired with a user and assigned to an object.

When setting permissions, verify all the object types are set with appropriate privileges for each particular action. Some operations require access permission at the root folder or parent folder in addition to access to the object being manipulated. Some operations require access or performance permission at a parent folder and a related object.

vCenter Server extensions might define additional privileges not listed here. Refer to the documentation for the extension for more information on those privileges.

Read the following topics next:

- Global Privileges
- Health Update Provider Privileges
- Host CIM Privileges
- Host Configuration Privileges
- Host Inventory
- Host Local Operations Privileges
- Host vSphere Replication Privileges
- Host Profile Privileges
- vSphere with Tanzu Privileges
- Network Privileges
- Performance Privileges
- Permissions Privileges
- Profile-driven Storage Privileges
- Resource Privileges
- Scheduled Task Privileges
- Sessions Privileges
- Storage Views Privileges
- Tasks Privileges
- Transfer Service Privileges
- VcTrusts/VcIdentity Privileges
- Trusted Infrastructure Administrator Privileges
- vApp Privileges
- VcIdentityProviders Privileges
- VMware vSphere Lifecycle Manager Configuration Privileges
- VMware vSphere Lifecycle Manager ESXi Health Perspectives Privileges
- VMware vSphere Lifecycle Manager General Privileges
- VMware vSphere Lifecycle Manager Hardware Compatibility Privileges
- VMware vSphere Lifecycle Manager Image Privileges
- VMware vSphere Lifecycle Manager Image Remediation Privileges
- VMware vSphere Lifecycle Manager Settings Privileges
- VMware vSphere Lifecycle Manager Manage Baseline Privileges
- VMware vSphere Lifecycle Manager Manage Patches and Upgrades Privileges

Alarms Privileges

Alarms privileges control the ability to create, modify, and respond to alarms on inventory objects.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-1. Alarms Privileges

Privilege Name	Description	Required On
Alarms.Acknowledge alarm	Allows suppression of all alarm actions on all triggered alarms.	Object on which an alarm is defined
Alarms.Create alarm	Allows creation of a new alarm. When creating alarms with a custom action, privilege to perform the action is verified when the user creates the alarm.	Object on which an alarm is defined
Alarms.Disable alarm action	Allows stopping an alarm action from occurring after an alarm has been triggered. This does not disable the alarm.	Object on which an alarm is defined
Alarms.Disable or enable alarm on entity	Allows enabling or disabling a particular alarm on a particular target type.	Object on which the alarm can trigger
Alarms.Modify alarm	Allows changing the properties of an alarm.	Object on which an alarm is defined

Table 16-1. Alarms Privileges (continued)

Privilege Name	Description	Required On
Alarms.Remove alarm	Allows deletion of an alarm.	Object on which an alarm is defined
Alarms.Set alarm status	Allows changing the status of the configured event alarm. The status can change to **Normal**, **Warning**, or **Alert**.	Object on which an alarm is defined

Auto Deploy and Image Profile Privileges

Auto Deploy privileges control who can perform different tasks on Auto Deploy rules, and who can associate a host. Auto Deploy privileges also allow you to control who can create or edit an image profile.

The table describes privileges that determine who can manage Auto Deploy rules and rule sets and who can create and edit image profiles. See *vCenter Server Installation and Setup*.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-2. Auto Deploy Privileges

Privilege Name	Description	Required On
Auto Deploy.Host.AssociateMachine	Allows users to associate a host with a machine.	vCenter Server
Auto Deploy.Image Profile.Create	Allows creation of image profiles.	vCenter Server
Auto Deploy.Image Profile.Edit	Allows editing of image profiles.	vCenter Server
Auto Deploy.Rule.Create	Allows creation of Auto Deploy rules.	vCenter Server
Auto Deploy.Rule.Delete	Allows deletion of Auto Deploy rules.	vCenter Server
Auto Deploy.Rule.Edit	Allows editing of Auto Deploy rules.	vCenter Server
Auto Deploy.RuleSet.Activate	Allows activation of Auto Deploy rule sets.	vCenter Server
Auto Deploy.RuleSet.Edit	Allows editing of Auto Deploy rule sets.	vCenter Server

Certificates Privileges

Certificates privileges control which users can manage ESXi certificates.

This privilege determines who can perform certificate management for ESXi hosts. See Required Privileges for Certificate Management Operations in the *vSphere Authentication* documentation for information on vCenter Server certificate management.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-3. Host Certificates Privileges

Privilege Name	Description	Required On
Certificates.Manage Certificates	Allows certificate management for ESXi hosts.	vCenter Server

Certificate Authority Privileges

Certificate authority privileges control aspects of VMware Certificate Authority (VMCA) certificates.

Table 16-4. Certificate Authority Privileges

Privilege Name	Description	Required On
Certificate Authority.Create/Delete (Admins priv).	Allows full administrative-level access for managing vCenter Server certificates.	vCenter Server
Certificate Authority.Create/Delete (below Admins priv).	Allows viewing the VMCA root certificate in the Certificate Management page in the vSphere Client.	vCenter Server

Certificate Management Privileges

Certificate management privileges control which users can manage vCenter Server certificates.

Table 16-5. Certificate Management Privileges

Privilege Name	Description	Required On
Certificate Management.Create/ Delete (Admins priv).	Allows full administrative-level access to various internal APIs and functionality for vCenter Server certificate-related operations.	vCenter Server
Certificate Management.Create/ Delete (below Admins priv).	Allows reduced administrative access to various internal APIs and functionality. This privilege restricts certificate related operations so that the user cannot escalate non-administrator privileges. Allowed operations are: ■ Generating certificate signing requests ■ Creating and retrieving Trusted Root chains ■ Deleting Trusted Root chains created by a user with the privilege **Certificate Management.Create/Delete (below Admins priv).** ■ Retrieving Machine SSL certificates ■ Retrieving the signing certificate chains for validating tokens issued by vCenter Server	vCenter Server

Cns Privileges

Cloud Native Store (Cns) privileges control which users can access the Cloud Native Storage UI.

Table 16-6. Cns Privileges

Privilege Name	Description	Required On
Cns.Searchable	Allows storage administrator to view the Cloud Native Storage UI.	Root vCenter Server

Content Library Privileges

Content Libraries provide simple and effective management for virtual machine templates and vApps. Content library privileges control who can view or manage different aspects of content libraries.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Note Inheritance of permissions for content libraries works in the context of a single vCenter Server instance. However, content libraries are not direct children of a vCenter Server system from an inventory perspective. The direct parent for content libraries is the global root. This means that if you set a permission at a vCenter Server level and propagate it to the children objects, the permission applies to data centers, folders, clusters, hosts, virtual machines, and so on, but does not apply to the content libraries that you see and operate with in this vCenter Server instance. To assign a permission on a content library, an Administrator must grant the permission to the user as a global permission. Global permissions support assigning privileges across solutions from a global root object.

Table 16-7. Content Library Privileges

Privilege Name	Description	Required On
Content library.Add library item	Allows addition of items in a library.	Library
Content library.Add root certificate to trust store	Allows addition of root certificates to the Trusted Root Certificates Store.	vCenter Server
Content library.Check in a template	Allows checking in of templates.	Library
Content library.Check out a template	Allows checking out of templates.	Library
Content library.Create a subscription for a published library	Allows creation of a library subscription.	Library
Content library.Create local library	Allows creation of local libraries on the specified vCenter Server system.	vCenter Server

Table 16-7. Content Library Privileges (continued)

Privilege Name	Description	Required On
Content library.Create or delete a Harbor registry	Allows creation or deletion of the VMware Tanzu Harbor Registry service.	vCenter Server for creation. Registry for deletion.
Content library.Create subscribed library	Allows creation of subscribed libraries.	vCenter Server
Content library.Create, delete or purge a Harbor registry project	Allows creation, deletion, or purging of VMware Tanzu Harbor Registry projects.	Registry
Content library.Delete library item	Allows deletion of library items.	Library. Set this permission to propagate to all library items.
Content library.Delete local library	Allows deletion of a local library.	Library
Content library.Delete root certificate from trust store	Allows deletion of root certificates from the Trusted Root Certificates Store.	vCenter Server
Content library.Delete subscribed library	Allows deletion of a subscribed library.	Library
Content library.Delete subscription of a published library	Allows deletion of a subscription to a library.	Library
Content library.Download files	Allows download of files from the content library.	Library
Content library.Evict library item	Allows eviction of items. The content of a subscribed library can be cached or not cached. If the content is cached, you can release a library item by evicting it if you have this privilege.	Library. Set this permission to propagate to all library items.
Content library.Evict subscribed library	Allows eviction of a subscribed library. The content of a subscribed library can be cached or not cached. If the content is cached, you can release a library by evicting it if you have this privilege.	Library
Content library.Import Storage	Allows a user to import a library item if the source file URL starts with `ds://` or `file://`. This privilege is disabled for content library administrator by default. Because an import from a storage URL implies import of content, enable this privilege only if necessary and if no security concern exists for the user who performs the import.	Library
Content library.Manage Harbor registry resources on specified compute resource	Allows management of VMware Tanzu Harbor Registry resources.	Compute cluster

Table 16-7. Content Library Privileges (continued)

Privilege Name	Description	Required On
Content library.Probe subscription information	This privilege allows solution users and APIs to probe a remote library's subscription info including URL, SSL certificate, and password. The resulting structure describes whether the subscription configuration is successful or whether there are problems such as SSL errors.	Library
Content library.Publish a library item to its subscribers	Allows publication of library items to subscribers.	Library. Set this permission to propagate to all library items.
Content library.Publish a library to its subscribers	Allows publication of libraries to subscribers.	Library
Content library.Read storage	Allows reading of content library storage.	Library
Content library.Sync library item	Allows synchronization of library items.	Library. Set this permission to propagate to all library items.
Content library.Sync subscribed library	Allows synchronization of subscribed libraries.	Library
Content library.Type introspection	Allows a solution user or API to introspect the type support plug-ins for the content library service.	Library
Content library.Update configuration settings	Allows you to update the configuration settings. No vSphere Client user interface elements are associated with this privilege.	Library
Content library.Update files	Allows you to upload content into the content library. Also allows you to remove files from a library item.	Library
Content library.Update library	Allows updates to the content library.	Library
Content library.Update library item	Allows updates to library items.	Library. Set this permission to propagate to all library items.
Content library.Update local library	Allows updates of local libraries.	Library
Content library.Update subscribed library	Allows you to update the properties of a subscribed library.	Library
Content library.Update subscription of a published library	Allows updates of subscription parameters. Users can update parameters such as the subscribed library's vCenter Server instance specification and placement of its virtual machine template items.	Library
Content library.View configuration settings	Allows you to view the configuration settings. No vSphere Client user interface elements are associated with this privilege.	Library

Cryptographic Operations Privileges

Cryptographic operations privileges control who can perform which type of cryptographic operation on which type of object.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-8. Cryptographic Operations Privileges

Privilege Name	Description	Required On
Cryptographic operations.Direct Access	Allows users access to encrypted resources. Users can export virtual machines, have NFC access to virtual machines, and open a console session to an encrypted virtual machine.	Virtual machine, host, or datastore
Cryptographic operations.Add disk	Allows users to add a disk to an encrypted virtual machine.	Virtual machine
Cryptographic operations.Clone	Allows users to clone an encrypted virtual machine.	Virtual machine
Cryptographic operations.Decrypt	Allows users to decrypt a virtual machine or disk.	Virtual machine
Cryptographic operations.Encrypt	Allows users to encrypt a virtual machine or a virtual machine disk.	Virtual machine
Cryptographic operations.Encrypt new	Allows users to encrypt a virtual machine during virtual machine creation or a disk during disk creation.	Virtual machine folder
Cryptographic operations.Manage encryption policies	Allows users to manage virtual machine storage policies with encryption IO filters. By default, virtual machines that use the Encryption storage policy do not use other storage policies.	vCenter Server root folder
Cryptographic operations.Manage KMS	Allows users to manage the Key Management Server for the vCenter Server system. Management tasks include adding and removing KMS instances, and establishing a trust relationship with the KMS.	vCenter Server system
Cryptographic operations.Manage keys	Allows users to perform key management operations. These operations are not supported from the vSphere Client but can be performed by using `crypto-util` or the API.	vCenter Server root folder

Table 16-8. Cryptographic Operations Privileges (continued)

Privilege Name	Description	Required On
Cryptographic operations.Migrate	Allows users to migrate an encrypted virtual machine to a different ESXi host. Supports migration with or without vMotion and storage vMotion. Supports migration to a different vCenter Server instance.	Virtual machine
Cryptographic operations.Recrypt	Allows users to recrypt virtual machines or disks with a different key. This privilege is required for both deep and shallow recrypt operations.	Virtual machine
Cryptographic operations.Register VM	Allows users to register an encrypted virtual machine with an ESXi host.	Virtual machine folder
Cryptographic operations.Register host	Allows users to enable encryption on a host. You can enable encryption on a host explicitly, or the virtual machine creation process can enable it.	Host folder for standalone hosts, cluster for hosts in cluster
Cryptographic operations.Read KMS information	Allows users to list vSphere Native Key Providers on the vCenter Server and on hosts. Also allows users to get vSphere Native Key Provider information.	vCenter Server or host

dvPort Group Privileges

Distributed virtual port group privileges control the ability to create, delete, and modify distributed virtual port groups.

The table describes the privileges required to create and configure distributed virtual port groups.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-9. Distributed Virtual Port Group Privileges

Privilege Name	Description	Required On
dvPort group.Create	Allows creation of a distributed virtual port group.	Virtual port groups
dvPort group.Delete	Allows deletion of distributed virtual port group. To have permission to perform this operation, a user or group must have this privilege assigned in both the object and its parent object.	Virtual port groups

Table 16-9. Distributed Virtual Port Group Privileges (continued)

Privilege Name	Description	Required On
dvPort group.Modify	Allows modification of a distributed virtual port group configuration.	Virtual port groups
dvPort group.Policy operation	Allows setting the policy of a distributed virtual port group.	Virtual port groups
dvPort group.Scope operation	Allows setting the scope of a distributed virtual port group.	Virtual port groups

Distributed Switch Privileges

Distributed Switch privileges control the ability to perform tasks related to the management of Distributed Switch instances.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-10. vSphere Distributed Switch Privileges

Privilege Name	Description	Required On
Distributed switch.Create	Allows creation of a distributed switch.	Data centers, Network folders
Distributed switch.Delete	Allows removal of a distributed switch. To have permission to perform this operation, a user or group must have this privilege assigned in both the object and its parent object.	Distributed switches
Distributed switch.Host operation	Allows changing the host members of a distributed switch.	Distributed switches
Distributed switch.Modify	Allows changing the configuration of a distributed switch.	Distributed switches
Distributed switch.Move	Allows moving a vSphere Distributed Switch to another folder.	Distributed switches
Distributed switch.Network I/O control operation	Allow changing the resource settings for a vSphere Distributed Switch.	Distributed switches
Distributed switch.Policy operation	Allows changing the policy of a vSphere Distributed Switch.	Distributed switches
Distributed switch .Port configuration operation	Allow changing the configuration of a port in a vSphere Distributed Switch.	Distributed switches
Distributed switch.Port setting operation	Allows changing the setting of a port in a vSphere Distributed Switch.	Distributed switches
Distributed switch.VSPAN operation	Allows changing the VSPAN configuration of a vSphere Distributed Switch.	Distributed switches

Datacenter Privileges

Datacenter privileges control the ability to create and edit data centers in the vSphere Client inventory.

All data center privileges are used in vCenter Server only. The **Create datacenter** privilege is defined on data center folders or the root object. All other data center privileges are pair with data centers, data center folders, or the root object.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-11. Datacenter Privileges

Privilege Name	Description	Required On
Datacenter.Create datacenter	Allows creation of new data center.	Data center folder or root object
Datacenter.Move datacenter	Allows moving a data center. Privilege must be present at both the source and destination.	Data center, source and destination
Datacenter.Network protocol profile configuration	Allows configuration of the network profile for a data center.	Data center
Datacenter.Query IP pool allocation	Allows configuration of a pool of IP addresses.	Data center
Datacenter.Reconfigure datacenter	Allows reconfiguration of a data center.	Data center
Datacenter.Release IP allocation	Allows releasing the assigned IP allocation for a data center.	Data center
Datacenter.Remove datacenter	Allows removal of a data center. To have permission to perform this operation, you must have this privilege assigned to both the object and its parent object.	Data center plus parent object
Datacenter.Rename datacenter	Allows changing the name of a data center.	Data center

Datastore Privileges

Datastore privileges control the ability to browse, manage, and allocate space on datastores.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-12. Datastore Privileges

Privilege Name	Description	Required On
Datastore.Allocate space	Allows allocating space on a datastore for a virtual machine, snapshot, clone, or virtual disk.	Data stores
Datastore.Browse datastore	Allows browsing files on a datastore.	Data stores
Datastore.Configure datastore	Allows configuration of a datastore.	Data stores
Datastore.Low level file operations	Allows performing read, write, delete, and rename operations in the datastore browser.	Data stores
Datastore.Move datastore	Allows moving a datastore between folders. Privileges must be present at both the source and destination.	Datastore, source and destination
Datastore.Remove datastore	Allows removal of a datastore. This privilege is deprecated. To have permission to perform this operation, a user or group must have this privilege assigned in both the object and its parent object.	Data stores
Datastore.Remove file	Allows deletion of files in the datastore. This privilege is deprecated. Assign the **Low level file operations** privilege.	Data stores
Datastore.Rename datastore	Allows renaming a datastore.	Data stores
Datastore.Update virtual machine files	Allows updating file paths to virtual machine files on a datastore after the datastore has been resignatured.	Data stores
Datastore.Update virtual machine metadata	Allows updating virtual machine metadata associated with a datastore.	Data stores

Datastore Cluster Privileges

Datastore cluster privileges control the configuration of datastore clusters for Storage DRS.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-13. Datastore Cluster Privileges

Privilege Name	Description	Required On
Datastore cluster.Configure a datatstore cluster	Allows creation of and configuration of settings for datastore clusters for Storage DRS.	Datastore clusters

ESX Agent Manager Privileges

ESX Agent Manager privileges control operations related to ESX Agent Manager and agent virtual machines. The ESX Agent Manager is a service that lets you install management virtual machines, which are tied to a host and not affected by VMware DRS or other services that migrate virtual machines.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-14. ESX Agent Manager

Privilege Name	Description	Required On
ESX Agent Manager.Config	Allows deployment of an agent virtual machine on a host or cluster.	Virtual machines
ESX Agent Manager.Modify	Allows modifications to an agent virtual machine such as powering off or deleting the virtual machine.	Virtual machines
ESX Agent View.View	Allows viewing of an agent virtual machine.	Virtual machines

Extension Privileges

Extension privileges control the ability to install and manage extensions.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-15. Extension Privileges

Privilege Name	Description	Required On
Extension.Register extension	Allows registration of an extension (plug-in).	Root vCenter Server
Extension.Unregister extension	Allows unregistering an extension (plug-in).	Root vCenter Server
Extension.Update extension	Allows updates to an extension (plug-in).	Root vCenter Server

External Stats Provider Privileges

External stats provider privileges control the ability to notify vCenter Server of Proactive Distributed Resource Scheduler (DRS) statistics.

These privileges apply to an API that is VMware-internal only.

Folder Privileges

Folder privileges control the ability to create and manage folders.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-16. Folder Privileges

Privilege Name	Description	Required On
Folder.Create folder	Allows creation of a new folder.	Folders
Folder.Delete folder	Allows deletion of a folder. To have permission to perform this operation, a user or group must have this privilege assigned in both the object and its parent object.	Folders
Folder.Move folder	Allows moving a folder. Privilege must be present at both the source and destination.	Folders
Folder.Rename folder	Allows changing the name of a folder.	Folders

Global Privileges

Global privileges control global tasks related to tasks, scripts, and extensions.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-17. Global Privileges

Privilege Name	Description	Required On
Global.Act as vCenter Server	Allows preparation or initiation of a vMotion send operation or a vMotion receive operation.	Root vCenter Server
Global.Cancel task	Allows cancellation of a running or queued task.	Inventory object related to the task
Global.Capacity planning	Allows enabling the use of capacity planning for planning consolidation of physical machines to virtual machines.	Root vCenter Server
Global.Diagnostics	Allows retrieval of a list of diagnostic files, log header, binary files, or diagnostic bundle. To avoid potential security breaches, limit this privilege to the vCenter Server Administrator role.	Root vCenter Server
Global.Disable methods	Allows servers for vCenter Server extensions to disable certain operations on objects managed by vCenter Server.	Root vCenter Server
Global.Enable methods	Allows servers for vCenter Server extensions to enable certain operations on objects managed byvCenter Server.	Root vCenter Server

Table 16-17. Global Privileges (continued)

Privilege Name	Description	Required On
Global.Global tag	Allows adding or removing global tags.	Root host or vCenter Server
Global.Health	Allows viewing the health of vCenter Server components.	Root vCenter Server
Global.Licenses	Allows viewing installed licenses and adding or removing licenses.	Root host or vCenter Server
Global.Log event	Allows logging a user-defined event against a particular managed entity.	Any object
Global.Manage custom attributes	Allows adding, removing, or renaming custom field definitions.	Root vCenter Server
Global.Proxy	Allows access to an internal interface for adding or removing endpoints to or from the proxy.	Root vCenter Server
Global.Script action	Allows scheduling a scripted action along with an alarm.	Any object
Global.Service managers	Allows use of the `resxtop` command in ESXCLI.	Root host or vCenter Server
Global.Set custom attribute	Allows viewing, creating, or removing custom attributes for a managed object.	Any object
Global.Settings	Allows reading and modifying runtime vCenter Server configuration settings.	Root vCenter Server
Global.System tag	Allows adding or removing system tags.	Root vCenter Server

Health Update Provider Privileges

Health update provider privileges control the ability for hardware vendors to notify vCenter Server of Proactive HA events.

These privileges apply to an API that is VMware-internal only.

Host CIM Privileges

Host CIM privileges control the use of CIM for host health monitoring.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-18. Host CIM Privileges

Privilege Name	Description	Required On
Host.CIM.CIM Interaction	Allow a client to obtain a ticket to use for CIM services.	Hosts

Host Configuration Privileges

Host configuration privileges control the ability to configure hosts.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-19. Host Configuration Privileges

Privilege Name	Description	Required On
Host.Configuration.Advanced Settings	Allows setting advanced host configuration options.	Hosts
Host.Configuration.Authentication Store	Allows configuring Active Directory authentication stores.	Hosts
Host.Configuration.Change PciPassthru settings	Allows changes to PciPassthru settings for a host.	Hosts
Host.Configuration.Change SNMP settings	Allows changes to SNMP settings for a host.	Hosts
Host.Configuration.Change date and time settings	Allows changes to date and time settings on the host.	Hosts
Host.Configuration.Change settings	Allows setting of lockdown mode on ESXi hosts.	Hosts
Host.Configuration.Connection	Allows changes to the connection status of a host (connected or disconnected).	Hosts
Host.Configuration.Firmware	Allows updates to the ESXi host's firmware.	Hosts
Host.Configuration.Hyperthreading	Allows enabling and disabling hyperthreading in a host CPU scheduler.	Hosts
Host.Configuration.Image configuration	Allows changes to the image associated with a host.	
Host.Configuration.Maintenance	Allows putting the host in and out of maintenance mode and shutting down and restarting the host.	Hosts
Host.Configuration.Memory configuration	Allows modifications to the host configuration.	Hosts
Host.Configuration.Network configuration	Allows configuration of network, firewall, and vMotion network.	Hosts
Host.Configuration.Power	Allows configuration of host power management settings.	Hosts
Host.Configuration.Query patch	Allows querying for installable patches and installing patches on the host.	Hosts
Host.Configuration.Security profile and firewall	Allows configuration of Internet services, such as SSH, Telnet, SNMP, and of the host firewall.	Hosts
Host.Configuration.Storage partition configuration	Allows VMFS datastore and diagnostic partition management. Users with this privilege can scan for new storage devices and manage iSCSI.	Hosts

Table 16-19. Host Configuration Privileges (continued)

Privilege Name	Description	Required On
Host.Configuration.System Management	Allows extensions to manipulate the file system on the host.	Hosts
Host.Configuration.System resources	Allows updates to the configuration of the system resource hierarchy.	Hosts
Host.Configuration.Virtual machine autostart configuration	Allows changes to the auto-start and auto-stop order of virtual machines on a single host.	Hosts

Host Inventory

Host inventory privileges control adding hosts to the inventory, adding hosts to clusters, and moving hosts in the inventory.

The table describes the privileges required to add and move hosts and clusters in the inventory.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-20. Host Inventory Privileges

Privilege Name	Description	Required On
Host.Inventory.Add host to cluster	Allows addition of a host to an existing cluster.	Clusters
Host.Inventory.Add standalone host	Allows addition of a standalone host.	Host folders
Host.Inventory.Create cluster	Allows creation of a new cluster.	Host folders
Host.Inventory.Modify cluster	Allows changing the properties of a cluster.	Clusters
Host.Inventory.Move cluster or standalone host	Allows moving a cluster or standalone host between folders. Privilege must be present at both the source and destination.	Clusters
Host.Inventory.Move host	Allows moving a set of existing hosts into or out of a cluster. Privilege must be present at both the source and destination.	Clusters
Host.Inventory.Remove cluster	Allows deletion of a cluster or standalone host. To have permission to perform this operation, a user or group must have this privilege assigned in both the object and its parent object.	Clusters, Hosts

Table 16-20. Host Inventory Privileges (continued)

Privilege Name	Description	Required On
Host.Inventory.Remove host	Allows removal of a host. To have permission to perform this operation, a user or group must have this privilege assigned in both the object and its parent object.	Hosts plus parent object
Host.Inventory.Rename cluster	Allows renaming a a cluster.	Clusters

Host Local Operations Privileges

Host local operations privileges control actions performed when the VMware Host Client is connected directly to a host.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-21. Host Local Operations Privileges

Privilege Name	Description	Required On
Host.Local operations.Add host to vCenter	Allows installation and removal of vCenter agents, such as vpxa and aam, on a host.	Root host
Host.Local operations.Create virtual machine	Allows creation of a new virtual machine from scratch on a disk without registering it on the host.	Root host
Host.Local operations.Delete virtual machine	Allows deletion of a virtual machine on disk. Supported for registered and unregistered virtual machines.	Root host
Host.Local operations.Manage user groups	Allows management of local accounts on a host.	Root host
Host.Local operations.Reconfigure virtual machine	Allows reconfiguring a virtual machine.	Root host

Host vSphere Replication Privileges

Host vSphere replication privileges control the use of virtual machine replication by VMware vCenter Site Recovery Manager™ for a host.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-22. Host vSphere Replication Privileges

Privilege Name	Description	Required On
Host.vSphere Replication.Manage Replication	Allows management of virtual machine replication on this host.	Hosts

Host Profile Privileges

Host Profile privileges control operations related to creating and modifying host profiles.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-23. Host Profile Privileges

Privilege Name	Description	Required On
Host profile.Clear	Allows clearing of profile related information.	Root vCenter Server
Host profile.Create	Allows creation of a host profile.	Root vCenter Server
Host profile.Delete	Allows deletion of a host profile.	Root vCenter Server
Host profile.Edit	Allows editing a host profile.	Root vCenter Server
Host profile.Export	Allows exporting a host profile	Root vCenter Server
Host profile.View	Allows viewing a host profile.	Root vCenter Server

vSphere with Tanzu Privileges

Namespaces privileges control who can create and manage VMware vSphere® with VMware Tanzu™ namespaces.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-24. Namespaces Privileges

Privilege Name	Description	Required On
Namespaces.Allows disk decommission operations	Allows for decommissioning operations of data stores.	Data stores
Namespaces.Backup Workloads component files	Allows for backing up the contents of the etcd cluster (used only in VMware Cloud on AWS).	Clusters
Namespaces.List Accessible Namespaces	Allows listing the accessible namespaces.	Clusters

Table 16-24. Namespaces Privileges (continued)

Privilege Name	Description	Required On
Namespaces.Modify cluster-wide configuration	Allows modifying the cluster-wide configuration, and enabling and disabling cluster namespaces.	Clusters
Namespaces.Modify cluster-wide namespace self-service configuration	Allows modifying the namespace self-service configuration.	Clusters (for activating and deactivating) Templates (for modifying the configuration) vCenter Server (for creating a template)
Namespaces.Modify namespace configuration	Allows modifying namespace configuration options such as resource allocation and user permissions.	Clusters
Namespaces.Toggle cluster capabilities	Allows manipulating the state of cluster capabilities (used internally only for VMware Cloud on AWS).	Clusters
Namespaces.Upgrade clusters to newer versions	Allows initiation of the cluster upgrade.	Clusters

Network Privileges

Network privileges control tasks related to network management.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-25. Network Privileges

Privilege Name	Description	Required On
Network.Assign network	Allows assigning a network to a virtual machine.	Networks, Virtual Machines
Network.Configure	Allows configuring a network.	Networks, Virtual Machines
Network.Move network	Allows moving a network between folders. Privilege must be present at both the source and destination.	Networks
Network.Remove	Allows removal of a network. This privilege is deprecated. To have permission to perform this operation, a user or group must have this privilege assigned in both the object and its parent object.	Networks

Performance Privileges

Performance privileges control modifying performance statistics settings.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-26. Performance Privileges

Privilege Name	Description	Required On
Performance.Modify intervals	Allows creating, removing, and updating performance data collection intervals.	Root vCenter Server

Permissions Privileges

Permissions privileges control the assigning of roles and permissions.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-27. Permissions Privileges

Privilege Name	Description	Required On
Permissions.Modify permission	Allows defining one or more permission rules on an entity, or updating rules if rules are already present for the given user or group on the entity. To have permission to perform this operation, a user or group must have this privilege assigned in both the object and its parent object.	Any object plus parent object
Permissions.Modify privilege	Allows modifying a privilege's group or description. No vSphere Client user interface elements are associated with this privilege.	
Permissions.Modify role	Allows updating a role's name and the privileges that are associated with the role.	Any object
Permissions.Reassign role permissions	Allows reassigning all permissions of a role to another role.	Any object

Profile-driven Storage Privileges

Profile-driven storage privileges control operations related to storage profiles.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-28. Profile-driven Storage Privileges

Privilege Name	Description	Required On
Profile-driven storage.Profile-driven storage update	Allows changes to be made to storage profiles, such as creating and updating storage capabilities and virtual machine storage profiles.	Root vCenter Server
Profile-driven storage.Profile-driven storage view	Allows viewing of defined storage capabilities and storage profiles.	Root vCenter Server

Resource Privileges

Resource privileges control the creation and management of resource pools, as well as the migration of virtual machines.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-29. Resource Privileges

Privilege Name	Description	Required On
Resource.Apply recommendation	Allows accepting a suggestion by the server to perform a migration with vMotion.	Clusters
Resource.Assign vApp to resource pool	Allows assignment of a vApp to a resource pool.	Resource pools
Resource.Assign virtual machine to resource pool	Allows assignment of a virtual machine to a resource pool.	Resource pools
Resource.Create resource pool	Allows creation of resource pools.	Resource pools, clusters
Resource.Migrate powered off virtual machine	Allows migration of a powered off virtual machine to a different resource pool or host.	Virtual machines
Resource.Migrate powered on virtual machine	Allows migration with vMotion of a powered on virtual machine to a different resource pool or host.	
Resource.Modify resource pool	Allows changes to the allocations of a resource pool.	Resource pools
Resource.Move resource pool	Allows moving a resource pool. Privilege must be present at both the source and destination.	Resource pools
Resource.Query vMotion	Allows querying the general vMotion compatibility of a virtual machine with a set of hosts.	Root vCenter Server

Table 16-29. Resource Privileges (continued)

Privilege Name	Description	Required On
Resource.Remove resource pool	Allows deletion of a resource pool. To have permission to perform this operation, a user or group must have this privilege assigned in both the object and its parent object.	Resource pools
Resource.Rename resource pool	Allows renaming of a resource pool.	Resource pools

Scheduled Task Privileges

Scheduled task privileges control creation, editing, and removal of scheduled tasks.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-30. Scheduled Task Privileges

Privilege Name	Description	Required On
Scheduled task.Create tasks	Allows scheduling of a task. Required in addition to the privileges to perform the scheduled action at the time of scheduling.	Any object
Scheduled task.Modify task	Allows reconfiguration of the scheduled task properties.	Any object
Scheduled task.Remove task	Allows removal of a scheduled task from the queue.	Any object
Scheduled task.Run task	Allows running the scheduled task immediately. Creating and running a scheduled task also requires permission to perform the associated action.	Any object

Sessions Privileges

Sessions privileges control the ability of extensions to open sessions on the vCenter Server system.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Note Assign Sessions privileges only to administrators or trusted users.

Table 16-31. Session Privileges

Privilege Name	Description	Required On
Sessions.Impersonate user	Allows impersonation of another user. This capability is used by extensions.	Root vCenter Server
Sessions.Message	Allows setting of the global login message.	Root vCenter Server
Sessions.Validate session	Allows verification of session validity.	Root vCenter Server
Sessions.View and stop sessions	Allows viewing sessions and forcing log out of one or more logged-on users.	Root vCenter Server

Storage Views Privileges

Storage Views privileges control privileges for Storage Monitoring Service APIs.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-32. Storage Views Privileges

Privilege Name	Description	Required On
Storage views.Configure service	Allows privileged users to use all Storage Monitoring Service APIs. Use **Storage views.View** for privileges to read-only Storage Monitoring Service APIs.	Root vCenter Server
Storage views.View	Allows privileged users to use read-only Storage Monitoring Service APIs.	Root vCenter Server

Tasks Privileges

Tasks privileges control the ability of extensions to create and update tasks on the vCenter Server.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-33. Tasks Privileges

Privilege Name	Description	Required On
Tasks.Create task	Allows an extension to create a user-defined task. No vSphere Client user interface elements are associated with this privilege.	Root vCenter Server
Tasks.Update task	Allows an extension to update a user-defined task. No vSphere Client user interface elements are associated with this privilege.	Root vCenter Server

Transfer Service Privileges

Transfer service privileges are VMware internal. Do not use these privileges.

VcTrusts/VcIdentity Privileges

VcTrusts/VcIdentity privileges control access to various internal APIs and functionality related to trust between vCenter Server systems.

Table 16-34. VcTrusts/VcIdentity Privileges

Privilege Name	Description	Required On
VcTrusts/VcIdentity.Create/Update/Delete (Admin privs)	Allows full administrative-level access to various internal APIs and functionality related to trust between vCenter Server systems.	N/A
VcTrusts/VcIdentity.Create/Update/Delete (below Admin privs)	Allows reduced administrative access to various internal APIs and functionality related to trust between vCenter Server systems. This privilege restricts creating/updating/deleting VcTrusts/VcIdentity so that the user cannot escalate non-administrator privileges.	N/A

Trusted Infrastructure Administrator Privileges

Trusted Infrastructure administrator privileges configure and manage a vSphere Trust Authority deployment.

These privileges determine who can perform configuration and management tasks for a vSphere Trust Authority deployment. See Prerequisites and Required Privileges for vSphere Trust Authority for more information about the Trust Authority roles and the TrustedAdmins group.

Table 16-35. Trusted Infrastructure Administrator Privileges

Privilege Name	Description	Required On
Trusted Infrastructure administrator.Configure Key Server Trust	Allows managing the Key Providers of the Key Provider Service.	Root vCenter Server
Trusted Infrastructure administrator.Configure Trust Authority Host TPM certificates	Allows creation and modification of the Attestation Service settings.	Root vCenter Server
Trusted Infrastructure administrator.Configure Trust Authority Host metadata	Allows editing the base images to be attested by the Attestation Service.	Root vCenter Server
Trusted Infrastructure administrator.Configure attesting SSO	Allows editing which hosts can be trusted by the Trust Authority Hosts.	Root vCenter Server

Table 16-35. Trusted Infrastructure Administrator Privileges (continued)

Privilege Name	Description	Required On
Trusted Infrastructure administrator.Configure token conversion policy	Allows configuring the token conversion policy.	Root vCenter Server
Trusted Infrastructure administrator.List Trusted Infrastructure Hosts	Allows reading information regarding the Trusted Hosts and the Trust Authority Hosts.	Root vCenter Server
Trusted Infrastructure administrator.List information about the STS	Allows exporting the Trusted Host details, so that they can be imported to the Trust Authority Cluster.	Root vCenter Server
Trusted Infrastructure administrator.Manage Trusted Infrastructure Hosts	Allows editing the information regarding the Trusted Hosts and the Trust Authority Hosts.	Root vCenter Server
Trusted Infrastructure administrator.Read Key Server Trust	Allows reading the Key Providers of the Key Provider Service.	Root vCenter Server
Trusted Infrastructure administrator.Read attesting SSO	Allows reading which hosts can be trusted by the Trust Authority Hosts.	Root vCenter Server
Trusted Infrastructure administrator.Retrieve TPM Trust Authority Host certificates	Allows reading the settings of the Attestation Service.	Root vCenter Server
Trusted Infrastructure administrator.Retrieve Trust Authority Host metadatata	Allows reading which base images can be attested by the Attestation Service.	Root vCenter Server

vApp Privileges

vApp privileges control operations related to deploying and configuring a vApp.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-36. vApp Privileges

Privilege Name	Description	Required On
vApp.Add virtual machine	Allows adding a virtual machine to a vApp.	vApps
vApp.Assign resource pool	Allows assigning a resource pool to a vApp.	vApps
vApp.Assign vApp	Allows assigning a vApp to another vApp	vApps
vApp.Clone	Allows cloning of a vApp.	vApps
vApp.Create	Allows creation of a vApp.	vApps

Table 16-36. vApp Privileges (continued)

Privilege Name	Description	Required On
vApp.Delete	Allows deletion a vApp. To have permission to perform this operation, a user or group must have this privilege assigned in both the object and its parent object.	vApps
vApp.Export	Allows export of a vApp from vSphere.	vApps
vApp.Import	Allows import of a vApp into vSphere.	vApps
vApp.Move	Allows moving a vApp to a new inventory location.	vApps
vApp.Power Off	Allows power off operations on a vApp.	vApps
vApp.Power On	Allows power on operations on a vApp.	vApps
vApp.Rename	Allows renaming a vApp.	vApps
vApp.Suspend	Allows suspension of a vApp.	vApps
vApp.Unregister	Allows unregistering a vApp. To have permission to perform this operation, a user or group must have this privilege assigned in both the object and its parent object.	vApps
vApp.View OVF Environment	Allows viewing the OVF environment of a powered-on virtual machine within a vApp.	vApps
vApp.vApp application configuration	Allows modification of a vApp's internal structure, such as product information and properties.	vApps
vApp.vApp instance configuration	Allows modification of a vApp's instance configuration, such as policies.	vApps
vApp.vApp managedBy configuration	Allows an extension or solution to mark a vApp as being managed by that extension or solution. No vSphere Client user interface elements are associated with this privilege.	vApps
vApp.vApp resource configuration	Allows modification of a vApp's resource configuration. To have permission to perform this operation, a user or group must have this privilege assigned in both the object and its parent object.	vApps

VcIdentityProviders Privileges

VcIdentityProviders privileges control access to the VcIdentityProviders API.

Table 16-37. VcIdentityProviders Privileges

Privilege Name	Description	Required On
VcIdentityProviders.Create	Allows create-only access to the VcIdentityProviders API (vCenter Server identity providers).	N/A
VcIdentityProviders.Manage	Allows administrative-level write access (create, read, update, delete) to the VcIdentityProviders API (vCenter Server identity providers).	N/A
VcIdentityProviders.Read	Allows read access to the VcIdentityProviders API (vCenter Server identity providers).	N/A

VMware vSphere Lifecycle Manager Configuration Privileges

VMware vSphere Lifecycle Manager configuration privileges control the ability to configure the vSphere Lifecycle Manager service.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Note Assign privileges that authorize users to invoke VMware vSphere Lifecycle Manager APIs that accept URLs only to administrators or trusted users.

Table 16-38. VMware vSphere Lifecycle Manager Configuration Privileges

Privilege Name	Description	Required On
VMware vSphere Lifecycle Manager.Configure.Configure Service	Allows configuring the vSphere Lifecycle Manager service and the scheduled patch download task.	Root vCenter Server

VMware vSphere Lifecycle Manager ESXi Health Perspectives Privileges

VMware vSphere Lifecycle Manager ESXi health perspective privileges control the ability to check the health of ESXi hosts and clusters.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-39. VMware vSphere Lifecycle Manager ESXi Health Perspectives Privileges

Privilege Name	Description	Required On
VMware vSphere Lifecycle Manager.ESXi Health Perspectives.Read	Allows querying the health of ESXi hosts and clusters.	Hosts Clusters
VMware vSphere Lifecycle Manager.ESXi Health Perspectives.Write	N/A	N/A

VMware vSphere Lifecycle Manager General Privileges

VMware vSphere Lifecycle Manager General privileges control the ability to read and write Lifecyle Manager resources.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-40. VMware vSphere Lifecycle Manager General Privileges

Privilege Name	Description	Required On
VMware vSphere Lifecycle Manager.Lifecycle Manager: General Privileges.Read	Allows reading of the vSphere Lifecycle Manager resources. This privilege is required to get task information.	Root vCenter Server
VMware vSphere Lifecycle Manager.Lifecycle Manager: General Privileges.Write	Allows writing of the vSphere Lifecycle Manager resources. This privilege is required to cancel a vSphere Lifecycle Manager task.	Root vCenter Server

VMware vSphere Lifecycle Manager Hardware Compatibility Privileges

VMware vSphere Lifecycle Manager Hardware Compatibility privileges control the ability to discover and resolve potential hardware compatibility issues.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-41. VMware vSphere Lifecycle Manager Hardware Compatibility Privileges

Privilege Name	Description	Required On
VMware vSphere Lifecycle Manager.Lifecycle Manager: Hardware Compatibility Privileges.Access Hardware Compatibility	Allows accessing the hardware compatibility data and resolving potential hardware compatibility issues.	Hosts

VMware vSphere Lifecycle Manager Image Privileges

VMware vSphere Lifecycle Manager Image privileges control the ability to manage images.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Note Assign privileges that authorize users to invoke VMware vSphere Lifecycle Manager APIs that accept URLs only to administrators or trusted users.

Table 16-42. VMware vSphere Lifecycle Manager Image Privileges

Privilege Name	Description	Required On
VMware vSphere Lifecycle Manager.Lifecycle Manager: Image Privileges.Read	Allows reading of vSphere Lifecycle Manager images. This privilege is required to: ■ List all the drafts for a cluster ■ Get more information on a draft ■ Perform a scan on a draft ■ Validate a draft ■ Retrieve the contents of a draft ■ Compute the effective component list ■ Get the contents of the current desired state document ■ Start a scan on a cluster ■ Get the compliance result ■ Get a recommendation ■ Export the current desired state as a depot, JSON file, or ISO	Root vCenter Server
VMware vSphere Lifecycle Manager.Lifecycle Manager: Image Privileges.Write	Allows managing of vSphere Lifecycle Manager images. This privilege is required to: ■ Create, delete, or commit a draft ■ Import the desired state ■ Generate recommendations ■ Set or delete different portions of a draft	Root vCenter Server

VMware vSphere Lifecycle Manager Image Remediation Privileges

VMware vSphere Lifecycle Manager Image privileges control the ability to remediate images.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-43. VMware vSphere Lifecycle Manager Image Remediation Privileges

Privilege Name	Description	Required On
VMware vSphere Lifecycle Manager.Lifecycle Manager: Image Remediation Privileges.Read	Allows performing the remediation pre-check.	Clusters
VMware vSphere Lifecycle Manager.Lifecycle Manager: Image Remediation Privileges.Write	Allows performing the remediation.	Clusters

VMware vSphere Lifecycle Manager Settings Privileges

VMware vSphere Lifecycle Manager Settings privileges control the ability to manage depots and remediation policies.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Note Assign privileges that authorize users to invoke VMware vSphere Lifecycle Manager APIs that accept URLs only to administrators or trusted users.

Table 16-44. VMware vSphere Lifecycle Manager Settings Privileges

Privilege Name	Description	Required On
VMware vSphere Lifecycle Manager.Lifecycle Manager: Settings Privileges.Read	Allows reading of vSphere Lifecycle Manager depots and remediation policies.	Root vCenter Server
VMware vSphere Lifecycle Manager.Lifecycle Manager: Settings Privileges.Write	Allows writing of vSphere Lifecycle Manager depots and remediation policies.	Root vCenter Server

VMware vSphere Lifecycle Manager Manage Baseline Privileges

VMware vSphere Lifecycle Manager Manage Baseline privileges control the ability to manage baselines and baseline groups.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-45. VMware vSphere Lifecycle Manager Manage Baseline Privileges

Privilege Name	Description	Required On
VMware vSphere Lifecycle Manager.Manage Baseline.Attach Baseline	Allows attaching baselines and baseline groups to objects in the vSphere inventory.	Root vCenter Server
VMware vSphere Lifecycle Manager.Manage Baseline.Manage Baseline	Allows creating, editing, or deleting baselines and baseline groups.	Root vCenter Server

VMware vSphere Lifecycle Manager Manage Patches and Upgrades Privileges

VMware vSphere Lifecycle Manager Manage Patches and Upgrades privileges control the ability to view, scan, and remediate applicable patches, extensions, or upgrades.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-46. VMware vSphere Lifecycle Manager Manage Patches and Upgrades Privileges

Privilege Name	Description	Required On
VMware vSphere Lifecycle Manager.Manage Patches and Upgrades.Remediate to Apply Patches, Extensions, and Upgrades	Allows remediation of virtual machines and hosts to apply patches, extensions, or upgrades when you are using baselines. In addition, this privilege allows viewing the compliance status.	Root vCenter Server
VMware vSphere Lifecycle Manager.Manage Patches and Upgrades.Scan for Applicable Patches, Extensions, and Upgrades	Allows scanning virtual machines and hosts to search for applicable patches, extensions, or upgrades when you are using baselines.	Root vCenter Server
VMware vSphere Lifecycle Manager.Manage Patches and Upgrades.Stage Patches and Extensions	Allows staging patches or extensions to ESXi hosts when you are using baselines. In addition, this privilege allows viewing the compliance status of ESXi hosts.	Root vCenter Server
VMware vSphere Lifecycle Manager.Manage Patches and Upgrades.View Compliance Status	Allows viewing the baseline compliance information for an object in the vSphere inventory.	Root vCenter Server

VMware vSphere Lifecycle Manager Upload File Privileges

VMware vSphere Lifecycle Manager Upload File privileges control the ability to import updates to the vSphere Lifecycle Manager depot.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Note Assign privileges that authorize users to invoke VMware vSphere Lifecycle Manager APIs that accept URLs only to administrators or trusted users.

Table 16-47. VMware vSphere Lifecycle Manager Upload File Privileges

Privilege Name	Description	Required On
VMware vSphere Lifecycle Manager.Upload file.Upload file	Allows uploading upgrade ISO and offline patch bundles.	Root vCenter Server

Virtual Machine Configuration Privileges

Virtual Machine Configuration privileges control the ability to configure virtual machine options and devices.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-48. Virtual Machine Configuration Privileges

Privilege Name	Description	Required On
Virtual machine.Configuration.Acquire disk lease	Allows disk lease operations for a virtual machine.	Virtual machines
Virtual machine.Configuration.Add existing disk	Allows adding an existing virtual disk to a virtual machine.	Virtual machines
Virtual machine.Configuration.Add new disk	Allows creation of a new virtual disk to add to a virtual machine.	Virtual machines
Virtual machine.Configuration.Add or remove device	Allows addition or removal of any non-disk device.	Virtual machines
Virtual machine.Configuration.Advanced configuration	Allows addition or modification of advanced parameters in the virtual machine's configuration file.	Virtual machines
Virtual machine.Configuration.Change CPU count	Allows changing the number of virtual CPUs.	Virtual machines
Virtual machine.Configuration.Change Memory	Allows changing the amount of memory allocated to the virtual machine.	Virtual machines
Virtual machine.Configuration.Change Settings	Allows changing general virtual machine settings.	Virtual machines
Virtual machine.Configuration.Change Swapfile placement	Allows changing the swapfile placement policy for a virtual machine.	Virtual machines
Virtual machine.Configuration.Change resource	Allows changing the resource configuration of a set of virtual machine nodes in a given resource pool.	Virtual machines

Table 16-48. Virtual Machine Configuration Privileges (continued)

Privilege Name	Description	Required On
Virtual machine.Configuration.Configure Host USB device	Allows attaching a host-based USB device to a virtual machine.	Virtual machines
Virtual machine.Configuration.Configure Raw device	Allows adding or removing a raw disk mapping or SCSI pass through device. Setting this parameter overrides any other privilege for modifying raw devices, including connection states.	Virtual machines
Virtual machine.Configuration.Configure managedBy	Allows an extension or solution to mark a virtual machine as being managed by that extension or solution.	Virtual machines
Virtual machine.Configuration.Display connection settings	Allows configuration of virtual machine remote console options.	Virtual machines
Virtual machine.Configuration.Extend virtual disk	Allows expansion of the size of a virtual disk.	Virtual machines
Virtual machine.Configuration.Modify device settings	Allows changing the properties of an existing device.	Virtual machines
Virtual machine.Configuration.Query Fault Tolerance compatibility	Allows checking if a virtual machine is compatible for Fault Tolerance.	Virtual machines
Virtual machine.Configuration.Query unowned files	Allows querying of unowned files.	Virtual machines
Virtual machine.Configuration.Reload from path	Allows changing a virtual machine configuration path while preserving the identity of the virtual machine. Solutions such as VMware vCenter Site Recovery Manager use this operation to maintain virtual machine identity during failover and failback.	Virtual machines
Virtual machine.Configuration.Remove disk	Allows removal of a virtual disk device.	Virtual machines
Virtual machine.Configuration.Rename	Allows renaming a virtual machine or modifying the associated notes of a virtual machine.	Virtual machines
Virtual machine.Configuration.Reset guest information	Allows editing the guest operating system information for a virtual machine.	Virtual machines
Virtual machine.Configuration.Set annotation	Allows adding or editing a virtual machine annotation.	Virtual machines

Table 16-48. Virtual Machine Configuration Privileges (continued)

Privilege Name	Description	Required On
Virtual machine.Configuration.Toggle disk change tracking	Allows enabling or disabling of change tracking for the virtual machine's disks.	Virtual machines
Virtual machine.Configuration.Toggle fork parent	Allows enabling or disabling a vmfork parent.	Virtual machines
Virtual machine.Configuration.Upgrade virtual machine compatibility	Allows upgrade of the virtual machine’s virtual machine compatibility version.	Virtual machines

Virtual Machine Guest Operations Privileges

Virtual Machine Guest Operations privileges control the ability to interact with files and applications inside a virtual machine's guest operating system with the API.

See the *vSphere Web Services API Reference* documentation for more information on these operations.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-49. Virtual Machine Guest Operations

Privilege Name	Description	Effective on Object
Virtual machine.Guest Operations.Guest Operation Alias modification	Allows virtual machine guest operations that involve modifying the alias for the virtual machine.	Virtual machines
Virtual machine.Guest Operations.Guest Operation Alias query	Allows virtual machine guest operations that involve querying the alias for the virtual machine.	Virtual machines
Virtual machine.Guest Operations.Guest Operation Modifications	Allows virtual machine guest operations that involve modifications to a guest operating system in a virtual machine, such as transferring a file to the virtual machine. No vSphere Client user interface elements are associated with this privilege.	Virtual machines

Table 16-49. Virtual Machine Guest Operations (continued)

Privilege Name	Description	Effective on Object
Virtual machine.Guest Operations.Guest Operation Program Execution	Allows virtual machine guest operations that involve running an application in the virtual machine. No vSphere Client user interface elements are associated with this privilege.	Virtual machines
Virtual machine.Guest Operations.Guest Operation Queries	Allows virtual machine guest operations that involve querying the guest operating system, such as listing files in the guest operating system. No vSphere Client user interface elements are associated with this privilege.	Virtual machines

Virtual Machine Interaction Privileges

Virtual Machine Interaction privileges control the ability to interact with a virtual machine console, configure media, perform power operations, and install VMware Tools.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-50. Virtual Machine Interaction

Privilege Name	Description	Required On
Virtual machine.Interaction.Answer question	Allows resolution of issues with virtual machine state transitions or runtime errors.	Virtual machines
Virtual machine.Interaction.Backup operation on virtual machine	Allows performance of backup operations on virtual machines.	Virtual machines
Virtual machine .Interaction.Configure CD media	Allows configuration of a virtual DVD or CD-ROM device.	Virtual machines
Virtual machine .Interaction.Configure floppy media	Allows configuration of a virtual floppy device.	Virtual machines

Table 16-50. Virtual Machine Interaction (continued)

Privilege Name	Description	Required On
Virtual machine.Interaction.Console interaction	Allows interaction with the virtual machine's virtual mouse, keyboard, and screen.	Virtual machines
Virtual machine.Interaction.Create screenshot	Allows creation of a virtual machine screen shot.	Virtual machines
Virtual machine.Interaction.Defragment all disks	Allows defragment operations on all disks of the virtual machine.	Virtual machines
Virtual machine.Interaction.Device connection	Allows changing the connected state of a virtual machine's disconnectable virtual devices.	Virtual machines
Virtual machine.Interaction.Drag and Drop	Allows drag and drop of files between a virtual machine and a remote client.	Virtual machines
Virtual machine.Interaction.Guest operating system management by VIX API	Allows management of the virtual machine's operating system through the VIX API.	Virtual machines
Virtual machine.Interaction.Inject USB HID scan codes	Allows injection of USB HID scan codes.	Virtual machines
Virtual machine.Interaction.Pause or Unpause	Allows pausing or unpausing of the virtual machine.	Virtual machines
Virtual machine.Interaction.Perform wipe or shrink operations	Allows performing wipe or shrink operations on the virtual machine.	Virtual machines
Virtual machine.Interaction.Power Off	Allows powering off a powered-on virtual machine. This operation powers down the guest operating system.	Virtual machines
Virtual machine.Interaction.Power On	Allows powering on a powered-off virtual machine, and resuming a suspended virtual machine.	Virtual machines
Virtual machine.Interaction.Record session on Virtual Machine	Allows recording a session on a virtual machine.	Virtual machines
Virtual machine.Interaction.Replay session on Virtual Machine	Allows replaying of a recorded session on a virtual machine.	Virtual machines

Table 16-50. Virtual Machine Interaction (continued)

Privilege Name	Description	Required On
Virtual machine.Interaction.Reset	Allows resetting of a virtual machine and reboots the guest operating system.	Virtual machines
Virtual machine.Interaction.Resume Fault Tolerance	Allows resuming of fault tolerance for a virtual machine.	Virtual machines
Virtual machine.Interaction.Suspend	Allows suspending a powered-on virtual machine. This operation puts the guest in standby mode.	Virtual machines
Virtual machine.Interaction.Suspend Fault Tolerance	Allows suspension of fault tolerance for a virtual machine.	Virtual machines
Virtual machine.Interaction.Test failover	Allows testing of Fault Tolerance failover by making the Secondary virtual machine the Primary virtual machine.	Virtual machines
Virtual machine.Interaction.Test restart Secondary VM	Allows termination of a Secondary virtual machine for a virtual machine using Fault Tolerance.	Virtual machines
Virtual machine.Interaction.Turn Off Fault Tolerance	Allows turning off Fault Tolerance for a virtual machine.	Virtual machines
Virtual machine.Interaction.Turn On Fault Tolerance	Allows turning on Fault Tolerance for a virtual machine.	Virtual machines
Virtual machine.Interaction.VMware Tools install	Allows mounting and unmounting the VMware Tools CD installer as a CD-ROM for the guest operating system.	Virtual machines

Virtual Machine Inventory Privileges

Virtual Machine Inventory privileges control adding, moving, and removing virtual machines.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-51. Virtual Machine Inventory Privileges

Privilege Name	Description	Required On
Virtual machine.Inventory.Create from existing	Allows creation of a virtual machine based on an existing virtual machine or template, by cloning or deploying from a template.	Clusters, Hosts, Virtual machine folders
Virtual machine.Inventory.Create new	Allows creation of a virtual machine and allocation of resources for its execution.	Clusters, Hosts, Virtual machine folders
Virtual machine.Inventory.Move	Allows relocating a virtual machine in the hierarchy. The privilege must be present at both the source and destination.	Virtual machines
Virtual machine.Inventory.Register	Allows adding an existing virtual machine to a vCenter Server or host inventory.	Clusters, Hosts, Virtual machine folders
Virtual machine.Inventory.Remove	Allows deletion of a virtual machine. Deletion removes the virtual machine's underlying files from disk. To have permission to perform this operation, a user or group must have this privilege assigned in both the object and its parent object.	Virtual machines
Virtual machine.Inventory.Unregister	Allows unregistering a virtual machine from a vCenter Server or host inventory. To have permission to perform this operation, a user or group must have this privilege assigned in both the object and its parent object.	Virtual machines

Virtual Machine Provisioning Privileges

Virtual Machine Provisioning privileges control activities related to deploying and customizing virtual machines.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-52. Virtual Machine Provisioning Privileges

Privilege Name	Description	Required On
Virtual machine.Provisioning.Allow disk access	Allows opening a disk on a virtual machine for random read and write access. Used mostly for remote disk mounting.	Virtual machines
Virtual machine.Provisioning.Allow file access	Allows operations on files associated with a virtual machine, including vmx, disks, logs, and nvram.	Virtual machines
Virtual machine.Provisioning.Allow read-only disk access	Allows opening a disk on a virtual machine for random read access. Used mostly for remote disk mounting.	Virtual machines

Table 16-52. Virtual Machine Provisioning Privileges (continued)

Privilege Name	Description	Required On
Virtual machine.Provisioning.Allow virtual machine download	Allows read operations on files associated with a virtual machine, including vmx, disks, logs, and nvram.	Root host or vCenter Server
Virtual machine.Provisioning.Allow virtual machine files upload	Allows write operations on files associated with a virtual machine, including vmx, disks, logs, and nvram.	Root host or vCenter Server
Virtual machine.Provisioning.Clone template	Allows cloning of a template.	Templates
Virtual machine.Provisioning.Clone virtual machine	Allows cloning of an existing virtual machine and allocation of resources.	Virtual machines
Virtual machine.Provisioning.Create template from virtual machine	Allows creation of a new template from a virtual machine.	Virtual machines
Virtual machine.Provisioning.Customize guest	Allows customization of a virtual machine's guest operating system without moving the virtual machine.	Virtual machines
Virtual machine.Provisioning.Deploy template	Allows deployment of a virtual machine from a template.	Templates
Virtual machine.Provisioning.Mark as template	Allows marking an existing powered off virtual machine as a template.	Virtual machines
Virtual machine.Provisioning.Mark as virtual machine	Allows marking an existing template as a virtual machine.	Templates
Virtual machine.Provisioning.Modify customization specification	Allows creation, modification, or deletion of customization specifications.	Root vCenter Server
Virtual machine.Provisioning.Promote disks	Allows promote operations on a virtual machine's disks.	Virtual machines
Virtual machine.Provisioning.Read customization specifications	Allows reading a customization specification.	Virtual machines

Virtual Machine Service Configuration Privileges

Virtual machine service configuration privileges control who can perform monitoring and management tasks on the service configuration.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-53. Virtual Machine Service Configuration Privileges

Privilege Name	Description
Virtual Machine.Service configuration.Allow notifications	Allows generating and consuming notification about service status.
Virtual Machine.Service configuration.Allow polling of global event notifications	Allows querying whether any notifications are present.
Virtual Machine.Service configuration.Manage service configurations	Allows creating, modifying, and deleting virtual machine services.
Virtual Machine.Service configuration.Modify service configuration	Allows modification of existing virtual machine service configuration.
Virtual Machine.Service configuration.Query service configurations	Allows retrieval of list of virtual machine services.
Virtual Machine.Service configuration.Read service configuration	Allows retrieval of existing virtual machine service configuration.

Virtual Machine Snapshot Management Privileges

Virtual machine snapshot management privileges control the ability to take, delete, rename, and restore snapshots.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-54. Virtual Machine State Privileges

Privilege Name	Description	Required On
Virtual machine.Snapshot management.Create snapshot	Allows creation of a snapshot from the virtual machine's current state.	Virtual machines
Virtual machine.Snapshot management.Remove Snapshot	Allows removal of a snapshot from the snapshot history.	Virtual machines

Table 16-54. Virtual Machine State Privileges (continued)

Privilege Name	Description	Required On
Virtual machine.Snapshot management.Rename Snapshot	Allows renaming a snapshot with a new name, a new description, or both.	Virtual machines
Virtual machine.Snapshot management.Revert to snapshot	Allows setting the virtual machine to the state it was in at a given snapshot.	Virtual machines

Virtual Machine vSphere Replication Privileges

Virtual Machine vSphere replication privileges control the use of replication by VMware vCenter Site Recovery Manager™ for virtual machines.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-55. Virtual Machine vSphere Replication

Privilege Name	Description	Required On
Virtual machine.vSphere Replication.Configure Replication	Allows configuration of replication for the virtual machine.	Virtual machines
Virtual machine.vSphere Replication.Manage Replication	Allows triggering of full sync, online sync or offline sync on a replication.	Virtual machines
Virtual machine .vSphere Replication.Monitor Replication	Allows monitoring of replication.	Virtual machines

vServices Privileges

vServices privileges control the ability to create, configure, and update vService dependencies for virtual machines and vApps.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-56. vServices

Privilege Name	Description	Required On
vService.Create dependency	Allows creation of a vService dependency for a virtual machine or vApp.	vApps and virtual machines
vService.Destroy dependency	Allows removal of a vService dependency for a virtual machine or vApp.	vApps and virtual machines
vService.Reconfigure dependency configuration	Allows reconfiguration of a dependency to update the provider or binding.	vApps and virtual machines
vService.Update dependency	Allows updates of a dependence to configure the name or description.	vApps and virtual machines

vSphere Tagging Privileges

vSphere Tagging privileges control the ability to create and delete tags and tag categories, and assign and remove tags on vCenter Server inventory objects.

You can set this privilege at different levels in the hierarchy. For example, if you set a privilege at the folder level, you can propagate the privilege to one or more objects within the folder. The object listed in the Required On column must have the privilege set, either directly or inherited.

Table 16-57. vSphere Tagging Privileges

Privilege Name	Description	Required On
vSphere Tagging.Assign or Unassign vSphere Tag	Allows assignment or unassignment of a tag for an object in the vCenter Server inventory.	Any object
vSphere Tagging.Assign or Unassign vSphere Tag on Object	Allows objects to have tags assigned or unassigned. Use this privilege to limit which objects users are able to assign or unassign tags to.	Any object
vSphere Tagging.Create vSphere Tag	Allows creation of a tag.	Any object
vSphere Tagging.Create vSphere Tag Category	Allows creation of a tag category.	Any object
vSphere Tagging.Delete vSphere Tag	Allows deletion of a tag.	Any object
vSphere Tagging.Delete vSphere Tag Category	Allows deletion of a tag category.	Any object
vSphere Tagging.Edit vSphere Tag	Allows editing of a tag.	Any object
vSphere Tagging.Edit vSphere Tag Category	Allows editing of a tag category.	Any object
vSphere Tagging.Modify UsedBy Field for Category	Allows changing the UsedBy field for a tag category.	Any object
vSphere Tagging.Modify UsedBy Field for Tag	Allows changing the UsedBy field for a tag.	Any object

vSphere Client Privileges

vSphere Client privileges control offline access to vCenter Server.

These privileges apply to VMware Cloud only.

Understanding vSphere Hardening and Compliance

17

Organizations expect to keep their data secure by reducing the risk of data theft, cyberattack, or unauthorized access. Organizations also must often comply with one or more regulations from government standards to private standards, such as the National Institute of Standards and Technology (NIST) and Defense Information Systems Agency Security Technical Implementation Guides (DISA STIG). Ensuring that your vSphere environment is in compliance with such standards involves understanding a broader set of considerations including people, processes, and technology.

A high-level overview of security and compliance topics that require attention helps you plan your compliance strategy. You also benefit from other compliance-related resources on the VMware Web site.

Read the following topics next:

Security Versus Compliance in the vSphere Environment

The terms security and compliance are often used interchangeably. However, they are unique and distinct concepts.

Security, often thought of as information security, is commonly defined as a set of technical, physical, and administrative controls that you implement to provide confidentiality, integrity, and availability. For example, you secure a host by locking down which accounts can log into it, and by what means (SSH, direct console, and so on). Compliance, by contrast, is a set

of requirements necessary to meet the minimum controls established by different regulatory frameworks that provide limited guidance on any specific type of technology, vendor, or configuration. For example, the Payment Card Industry (PCI) has established security guidelines to help organizations proactively protect customer account data.

Security reduces the risk of data theft, cyberattack, or unauthorized access, while compliance is the proof that a security control is in place, typically within a defined time line. Security is primarily outlined in the design decisions and highlighted within the technology configurations. Compliance is focused on mapping the correlation between security controls and specific requirements. A compliance mapping provides a centralized view to list out many of the required security controls. Those controls are further detailed by including each security control's respective compliance citations as dictated by a domain such as NIST, PCI, FedRAMP, HIPAA, and so forth.

Effective cybersecurity and compliance programs are built on three pillars: people, process, and technology. A general misconception is that technology alone can solve all your cybersecurity needs. Technology does play a large and important role in the development and execution of an information security program. However, technology without process and procedures, awareness and training, creates a vulnerability within your organization.

When defining your security and compliance strategies, keep the following in mind:

- People need general awareness and training, whereas IT staff need specific training.
- Process defines how your organization's activities, roles, and documentation are used to mitigate risk. Processes are only effective if people follow them correctly.
- Technology can be used to prevent or reduce the impact of cybersecurity risk to your organization. Which technology to use depends on your organization's risk acceptance level.

VMware provides Compliance Kits that contain both an Audit Guide and a Product Applicability Guide, helping to bridge the gap between compliance and regulatory requirements and implementation guides. For more information, see https://core.vmware.com/compliance.

Glossary of Compliance Terms

Compliance introduces specific terms and definitions that are important to understand.

Table 17-1. Compliance Terms

Term	Definition
CJIS	Criminal Justice Information Services. In the context of compliance, the CJIS produces a Security Policy for how local, state, and federal criminal justice and law enforcement agencies must take security precautions to protect sensitive information such as fingerprints and criminal backgrounds.
DISA STIG	Defense Information Systems Agency Security Technical Implementation Guide. The Defense Information Systems Agency (DISA) is the entity responsible for maintaining the security posture of the Department of Defense (DoD) IT infrastructure. DISA accomplishes this task by developing and using Security Technical Implementation Guides, or "STIGs."
FedRAMP	Federal Risk and Authorization Management Program. FedRAMP is a government-wide program that provides a standardized approach to security assessment, authorization, and continuous monitoring for cloud products and services.
HIPAA	Health Insurance Portability and Accountability Act. Passed by Congress in 1996, HIPAA does the following: ■ Gives millions of American workers and their families the ability to transfer and continue health insurance coverage for when they change or lose jobs ■ Reduces health care fraud and abuse ■ Mandates industry-wide standards for health care information on electronic billing and other processes ■ Requires the protection and confidential handling of protected health information The latter bullet is of most importance to *vSphere Security* documentation.
NCCoE	National Cybersecurity Center of Excellence. NCCoE is a U.S government organization that produces and publicly shares solutions to cybersecurity problems that U.S. businesses encounter. The center forms a team of people from cybersecurity technology companies, other federal agencies, and academia to address each problem.
NIST	National Institute of Standards and Technology. Founded in 1901, NIST is a non-regulatory federal agency within the U.S. Department of Commerce. NIST's mission is to advocate for U.S. innovation and industrial competitiveness by advancing measurement science, standards, and technology in ways that increase economic security and improve our quality of life.
PAG	Product Applicability Guide. A document that provides general guidance for organizations that are considering a company's solutions to help them address compliance requirements.

Table 17-1. Compliance Terms (continued)

Term	Definition
PCI DSS	Payment Card Industry Data Security Standard. A set of security standards designed to ensure that all companies that accept, process, store, or transmit credit card information maintain a secure environment.
VVD/VCF Compliance Solutions	VMware Validated Design/VMware Cloud Foundation. The VMware Validated Designs provide comprehensive and extensively tested blueprints to build and operate a Software-Defined Data Center. VVD/VCF compliance solutions enable customers to meet compliance requirements for multiple government and industry regulations.

Understanding the vSphere Security Configuration Guide

VMware creates Security Hardening Guides that provide prescriptive guidance about deploying and operating VMware products in a secure manner. For vSphere, this guide is called the *vSphere Security Configuration Guide* (formerly known as the *Hardening Guide*).

The *vSphere Security Configuration Guide* contains security best practices for vSphere. The *vSphere Security Configuration Guide* does not map directly to regulatory guidelines or frameworks, and so is not a compliance guide. Also, the *vSphere Security Configuration Guide* is not intended for use as a security checklist. Security is always a tradeoff. When you implement security controls, you might affect usability, performance, or other operational tasks negatively. Consider your workloads, usage patterns, organizational structure, and so on carefully before making security changes, whether the advice is from VMware or from other industry sources. If your organization is subject to regulatory compliance needs, see Security Versus Compliance in the vSphere Environment or visit https://core.vmware.com/compliance. This site features compliance kits and product audit guides to help vSphere administrators and regulatory auditors secure and attest virtual infrastructure for regulatory frameworks, such as NIST 800-53v4, NIST 800-171, PCI DSS, HIPAA, CJIS, ISO 27001, and more.

The *vSphere Security Configuration Guide* does not discuss securing the following items:

- Software running inside the virtual machine, such as the Guest OS and applications
- Traffic running through the virtual machine networks
- Security of add-on products

The *vSphere Security Configuration Guide* is not meant to be used as a "compliance" tool. The *vSphere Security Configuration Guide* does enable you to take initial steps towards compliance, but used by itself, it does not ensure that your deployment is compliant. For more information about compliance, see Security Versus Compliance in the vSphere Environment.

Reading the vSphere Security Configuration Guide

The *vSphere Security Configuration Guide* is a spreadsheet that contains security-related guidelines to assist you with modifying your vSphere security configuration. These guidelines are group into tabs based on the affected components, with some or all of the following columns.

Table 17-2. vSphere Security Configuration Guide Spreadsheet Columns

Column Heading	Description
Guideline ID	A unique two-part ID to reference a security configuration or hardening recommendation. The first part indicates the component, defined as follows: ■ ESXi: ESXi hosts ■ VM: Virtual machines ■ vNetwork: Virtual switches
Description	A short explanation of the particular recommendation.
Discussion	Description of the vulnerability behind a particular recommendation.
Configuration Parameter	Provides the applicable configuration parameter or filename, if any.
Desired Value	The desired state or value of the recommendation. Possible values include: ■ N/A ■ Site Specific ■ False ■ True ■ Enabled ■ Disabled ■ Not present or False
Default Value	The default value set by vSphere.
Is desired value the default?	States if the security setting is the default product configuration.
Action Needed	The type of action to take on the particular recommendation. Actions include: ■ Update ■ Audit Only ■ Modify ■ Add ■ Remove
Setting Location in the vSphere Client	Steps for checking on the value by using the vSphere Client.
Negative Functional Impact in Change From Default?	Description, if any, of a potential negative impact from using the security recommendation.
PowerCLI Command Assessment	Steps for checking on the value by using PowerCLI.
PowerCLI Command Remediation Example	Steps for setting (remediating) the value by using PowerCLI.
vCLI Command Remediation	Steps for setting (remediating) the value by using the vCLI commands.
PowerCLI Command Assessment	Steps for checking on the value by using the PowerCLI commands.

Table 17-2. vSphere Security Configuration Guide Spreadsheet Columns (continued)

Column Heading	Description
PowerCLI Command Remediation	Steps for setting (remediating) the value by using the PowerCLI commands.
Able to set using Host Profile	Whether the setting can be accomplished by using Host Profiles (applies only to ESXi guidelines).
Hardening	If TRUE, then the guideline has only one implementation to be compliant. If FALSE then you can satisfy the guideline implementation by more than one configuration setting. The actual setting is often site-specific.
Site Specific Setting	If TRUE, then the setting to be compliant with the guideline depends on rules or standards that are specific to that vSphere deployment.
Audit Setting	If TRUE, then the value of the listed setting might need to be modified to satisfy site-specific rules.

Note These columns might change over time as required. For example, recent additions include the DISA STIG ID, Hardening, and Site Specific Setting columns. Check https://blogs.vmware.com for announcements about updates to the *vSphere Secure Configuration Guide*.

Do not blindly apply guidelines in the *vSphere Secure Configuration Guide* to your environment. Rather, take time to evaluate each setting and make an informed decision whether you want to apply it. At a minimum, you can use the instructions in the Assessment columns to verify the security of your deployment.

The *vSphere Secure Configuration Guide* is an aid to begin implementing compliance in your deployment. When used with the Defense Information Systems Agency (DISA) and other compliance guidelines, the *vSphere Secure Configuration Guide* enables you to map vSphere security controls to the compliance flavor per each guideline.

About the National Institute of Standards and Technology

The National Institute of Standards and Technology (NIST) is a non-regulatory government agency that develops technology, metrics, standards, and guidelines. Compliance with NIST standards and guidelines has become a top priority in many industries today.

The National Institute of Standards and Technology (NIST) was founded in 1901 and is now part of the U.S. Department of Commerce. NIST is one of the nation's oldest physical science laboratories. Today, NIST measurements support the smallest of technologies to the largest and most complex of human-made creations, from nanoscale devices, up to earthquake-resistant skyscrapers and global communication networks.

The Federal Information Security Management Act (FISMA) is a United States federal law passed in 2002 that made it a requirement for federal agencies to develop, document, and implement an information security and protection program. NIST plays an important role in the FISMA implementation by producing key security standards and guidelines (for example, FIPS 199, FIPS 200, and SP 800 series).

Government and private organizations use NIST 800-53 to secure information systems. Cybersecurity and privacy controls are essential to protect organizational operations (including mission, functions, image, and reputation), organizational assets, and individuals from a diverse set of threats. Some of these threats include hostile cyber-attacks, natural disasters, structural failures, and human errors. VMware has enlisted a third-party audit partner to evaluate VMware products and solutions against the NIST 800-53 catalog of controls. For more information, visit the NIST webpage at https://www.nist.gov/cyberframework.

About DISA STIGs

The Defense Information Systems Agency (DISA) develops and publishes Security Technical Implementation Guides, or "STIGs." DISA STIGs provide technical guidance for hardening systems and reducing threats.

The Defense Information Systems Agency (DISA) is the U.S. Department of Defense (DoD) combat support agency responsible for maintaining the security posture of the DOD Information Network (DODIN). One of the ways DISA accomplishes this task is by developing, disseminating, and mandating the implementation of Security Technical Implementation Guides, or STIGs. In brief, STIGs are portable, standards-based guides for hardening systems. STIGs are mandatory for U.S. DoD IT systems and, as such, provide a vetted, secure baseline for non-DoD entities to measure their security posture.

Vendors such as VMware submit suggested security hardening guidance to DISA for evaluation, based on DISA protocols and feedback. Once that process is complete, the official STIG is published on the DISA organization's web site at https://public.cyber.mil/stigs/. VMware provides security baselines and hardening guidance for vSphere as part of the *vSphere Security Configuration Guide*. See https://core.vmware.com/security.

About VMware Security Development Lifecycle

The VMware Security Development Lifecycle (SDL) program identifies and mitigates security risk during the development phase of VMware software products. VMware also operates the VMware Security Response Center (VSRC) to conduct the analysis and remediation of software security issues in VMware products.

The SDL is the software development methodology that the VMware Security Engineering, Communication, and Response (vSECR) group, and VMware product development groups, use to help identify and mitigate security issues. For more information about the VMware Security Development Lifecycle, see the webpage at https://www.vmware.com/security/sdl.html.

The VSRC works with customers and the security research community to achieve the goals of addressing security issues and providing customers with actionable security information in a timely manner. For more information about the VMware Security Response Center, see the webpage at https://www.vmware.com/security/vsrc.html.

Audit Logging

Audit logging of network traffic, compliance alerts, firewall activity, operating system changes, and provisioning activities is considered a best practice for maintaining the security of any IT environment. In addition, logging is a specific requirement of many regulations and standards.

One of the first steps to take for ensuring you are aware of changes to your infrastructure is to audit your environment. By default, vSphere includes tools that enable you to view and track changes. For example, you can use the Tasks and Events tab in the vSphere Client on any object in your vSphere hierarchy to see what changes have occurred. You can also use the PowerCLI to retrieve events and tasks. Also, vRealize Log Insight offers audit logging to support collection and retention of important system events. In addition, many third-party tools are available that provide vCenter auditing.

Log files can provide an audit trail to help determine who or what is accessing a host, a virtual machine, and so on. For more information, see ESXi Log File Locations.

Single Sign-On Audit Events

Single Sign-On (SSO) audit events are records of user or system actions for accessing the SSO services.

vCenter Server 6.7 Update 2 and later improves VMware vCenter Single Sign-On auditing by adding events for the following operations:

- User management
- Login
- Group creation
- Identity source
- Policy updates

The supported identity sources are vsphere.local, Integrated Windows Authentication (IWA), and Active Directory over LDAP.

When a user logs in to vCenter Server through Single Sign-On, or makes changes that affect SSO, the following audit events are written to the SSO audit log file:

- **Login and Logout Attempts:** Events for all the successful and failed login and logout operations.
- **Privilege Change:** Event for change in a user role or permissions.
- **Account Change:** Event for change in the user account information, for example, user name, password, or any additional account information.
- **Security Change:** Event for change in a security configuration, parameter, or policy.
- **Account Enabled or Disabled:** Event for when an account is enabled or disabled.
- **Identity Source:** Event for adding, deleting, or editing an identity source.

In the vSphere Client, event data is displayed in the **Monitor** tab. See the *vSphere Monitoring and Performance* documentation.

SSO audit event data includes the following details:

- Timestamp of when the event occurred.
- User who performed the action.
- Description of the event.
- Severity of the event.
- IP address of client used to connect to vCenter Server, if available.

SSO Audit Event Log Overview

The vSphere Single-Sign On process writes audit events to the `audit_events.log` file in the `/var/log/audit/sso-events/` directory.

Caution Never manually edit the `audit_events.log` file, as doing so might cause the audit logging to fail.

Keep the following in mind when working with the `audit_events.log` file:

- The log file is archived once it reaches 50 MB.
- A maximum of 10 archive files is kept. If the limit is reached, the oldest file is purged when a new archive is created.
- The archive files are named `audit_events-<index>.log.gz`, where the index is a numeral from 1 to 10. The first archive created is index 1, and is increased with each subsequent archive.
- The oldest events are in archive index 1. The highest indexed file is the latest archive.

Understanding Security and Compliance Next Steps

Conducting a security assessment is the first step in understanding any vulnerabilities in your infrastructure. A security assessment is part of a security audit, which looks at both systems and practices, including security compliance.

A security assessment generally refers to scanning your organization's physical infrastructure (firewalls, networks, hardware, and so on) to identify vulnerabilities and flaws. A security assessment is not the same as a security audit. A security audit includes not only a review of physical infrastructure but other areas such as policy and standard operating procedures, including security compliance. After you have the audit, you can decide on the steps to remedy the problems within the system.

You might ask these general questions when preparing to conduct a security audit:

1 Is our organization mandated to adhere to a compliance regulation? If so which one(s)?

2 What is our audit interval?

3 What is our internal self-assessment interval?

4 Do we have access to previous audit results and have we viewed them?

5 Do we use a third-party audit firm to help us prepare for an audit? If so, what is their level of comfort with virtualization?

6 Do we run vulnerability scans against the systems and applications? When and how often?

7 What are our internal cybersecurity policies?

8 Is your audit logging configured according to your needs? See Audit Logging.

In the absence of specific guidance or direction on where to begin, you can jumpstart securing your vSphere environment by:

- Keeping your environment up-to-date with the latest software and firmware patches
- Maintaining good password management and hygiene for all accounts
- Reviewing vendor-approved security recommendations
- Referring to the VMware Security Configuration Guides (see Understanding the vSphere Security Configuration Guide)
- Using readily available and proven guidance from policy frameworks such as NIST, ISO, and so forth
- Following guidance from regulatory compliance frameworks such as PCI, DISA, and FedRAMP

vCenter Server and FIPS

In vSphere 7.0 Update 2 and later, you can enable FIPS-validated cryptography on the vCenter Server Appliance.

FIPS 140-2 is a U.S. and Canadian government standard that specifies security requirements for cryptographic modules. vSphere uses FIPS-validated cryptographic modules to match those specified by the FIPS 140-2 standard. The goal of vSphere FIPS support is to ease the compliance and security activities in various regulated environments.

In vSphere 6.7 and later, ESXi and vCenter Server use FIPS-validated cryptography to protect management interfaces and the VMware Certificate Authority (VMCA).

vSphere 7.0 Update 2 and later adds additional FIPS-validated cryptography to vCenter Server Appliance. By default, this FIPS validation option is disabled.

Note vSphere favors compatibility over FIPS, so some components have considerations to be aware of. See Considerations When Using FIPS.

FIPS Modules

A cryptographic module is a set of hardware, software, or firmware that implements security functions. ESXi uses several FIPS 140-2 validated cryptographic modules.

The following table shows the set of FIPS 140-2 validated cryptographic modules in use by ESXi.

Table 17-3. FIPS Modules

Cryptographic Module	Security Policy Version	Algorithms (CAVP)	Cryptographic Module Validation Program
Vmkernel Cryptographic Module	1.0	AES, SHS, DRBG, HMAC (C 1172)	Certificate #3073
Vmkernel Cryptographic Module Loader	Not applicable	HMAC, SHS (C 1171)	Certificate #3073
Vmkernel DRBG Cryptographic Module	Not applicable	AES, DRBG (C 499)	NA
VMware OpenSSL FIPS Object Module	2.0.20-vmw	DRBG, AES, SHS, HMAC, DSA, RSA, ECDSA, KAS-FFC, KAS-ECC (C 470)	Certificate #3550 and #3857

Enable and Disable FIPS on the vCenter Server Appliance

You can enable or disable FIPS-validated cryptography on the vCenter Server Appliance by using HTTP requests.

You can use various ways to execute HTTP requests. This task shows how to use the Developer Center in the vSphere Client to enable and disable FIPS on the vCenter Server Appliance. See *VMware vCenter Server Management Programming Guide* for more information about using APIs to work with the vCenter Server Appliance.

Procedure

1. Log in to the vCenter Server system with the vSphere Client.
2. From the Menu, select **Developer Center**.
3. Click **API Explorer**.
4. From the **Select API** drop-down menu, select **appliance**.
5. Scroll down through the categories and expand **system/security/global_fips**.
6. Expand **GET** and click **Execute** under **Try it out**.

   You can view the current setting under **Response**.

7 Change the setting.

a To enable FIPS, expand **PUT**, enter the following in the `request_body`, and click **Execute**.

```
{
"enabled":true
}
```

b To disable FIPS, expand **PUT**, enter the following in the `request_body`, and click **Execute**.

```
{
"enabled":false
}
```

Results

The vCenter Server Appliance reboots after you enable or disable FIPS.

Considerations When Using FIPS

When enabling FIPS on vCenter Server Appliance, some components present functional constraints currently.

You should see no differences after enabling FIPS on vCenter Server, however there are some considerations to be aware of.

Table 17-4. FIPS Considerations

Product or Component	Consideration	Workaround
vSphere Single Sign-On	When you enable FIPS, vCenter Server supports only cryptographic modules for federated authentication. As a result, RSA SecureID and some CAC cards no longer function.	Use federated authentication. See the *vSphere Authentication* documentation for details.
Non-VMware and partner vSphere Client UI plug-ins	These plug-ins might not work with FIPS enabled.	Upgrade plug-ins to use conformant encryption libraries. See "Preparing Local Plug-ins for FIPS Compliance" at https://code.vmware.com/docs/13385/preparing-local-plug-ins-for-fips-compliance.
vCenter Server file-based backup and restore mechanism	File-based backup and restore with SMB is not FIPS compliant.	Use a different protocol for backup and restore (FTP, FTPS, HTTP, HTTPS, SFTP, or NFS).